FRONTEND_GUIDELINES.md — VERDICT

Design System & Frontend Standards
B2B Deposition Coaching Platform for Litigation Attorneys
Version: 1.0.0 | February 21, 2026

TABLE OF CONTENTS

---

1. DESIGN PRINCIPLES

VERDICT is used by litigation partners and senior associates — people who bill $1,000/hour and have zero tolerance for ambiguity. Every design decision must serve the following principles:

1.1 Command Clarity

Every screen communicates exactly one primary action. Attorneys should never scan the UI wondering what to do next. In the live session screen, the alert rail is the only thing that moves — everything else is stable.

> **Rule:** One primary CTA per screen. All other actions are secondary or tertiary. Never two blue buttons on the same screen.

1.2 Information Density Without Noise

The three-panel live session layout shows transcript, controls, and alerts simultaneously — but each panel is self-contained. High density serves experienced users; visual noise is the enemy.

> **Rule:** Use whitespace intentionally. Pack information in cards with clear hierarchy, never in raw prose.

1.3 Status Always Visible

Agents are running, documents are indexing, timers are counting down. Attorneys need ambient status without modal interruptions. Status lives in badges, rail headers, and subtle indicators — never in popups.

> **Rule:** Async state is always represented. A silent spinner is acceptable; a blank element while loading is not.

1.4 Trust Through Polish

A law firm will not trust a tool that looks like a demo. Every interaction must feel deliberate — transitions are smooth, alerts have weight, scores feel earned. Visual quality communicates product reliability.

Rule: Animations are purposeful, not decorative. Every motion communicates a state change. Nothing bounces without cause.

1.5 Accessibility is Non-Negotiable

Enterprise legal tools face accessibility audits. VERDICT targets WCAG 2.1 AA compliance across all P0 screens.

Rule: Every component ships with keyboard navigation, ARIA labels, and 4.5:1 contrast on all text. No exceptions.

---

2. DESIGN TOKENS

2.1 Color Palette

Primary — Verdict Navy (Dark UI Foundation)

```
--verdict-navy-900: #0A0F1C;   /* darkest — page bg outside panels */
--verdict-navy-800: #0F1729;   /* PRIMARY PAGE BACKGROUND ✦ */
--verdict-navy-700: #131E30;   /* subtle depth, left/right panels */
--verdict-navy-600: #1E2B3C;   /* card/panel backgrounds ✦ */
--verdict-navy-500: #253347;   /* elevated cards, modals */
--verdict-navy-400: #2D3E54;   /* hover states on panels */
--verdict-navy-300: #374B63;   /* borders, dividers */
--verdict-navy-200: #4A6080;   /* muted text, placeholders */
--verdict-navy-100: #6B849E;   /* secondary text */
--verdict-navy-50:  #94A3B8;   /* tertiary text, captions */
```

Blue — Primary Accent (Attorney Actions)

```
--verdict-blue-900: #1E3A5F;   /* deep press state */
--verdict-blue-800: #1D4ED8;   /* active links */
--verdict-blue-700: #2563EB;   /* primary button hover */
--verdict-blue-600: #3B82F6;   /* PRIMARY BUTTON ✦ */
--verdict-blue-500: #60A5FA;   /* icon fills, active tabs */
--verdict-blue-400: #93C5FD;   /* disabled state */
--verdict-blue-300: #BFDBFE;   /* light accents */
--verdict-blue-100: #DBEAFE;   /* subtle highlight bg */
--verdict-blue-50:  #EFF6FF;   /* info backgrounds */
```

Red — Alerts & Risk (Inconsistency / HIGH IMPEACHMENT)

```
--verdict-red-900: #7F1D1D;   /* deep danger */
--verdict-red-800: #991B1B;   /* danger hover */
--verdict-red-700: #B91C1C;   /* HIGH IMPEACHMENT RISK badge bg */
--verdict-red-600: #DC2626;   /* danger button */
--verdict-red-500: #EF4444;   /* PRIMARY ALERT COLOR ✦ */
--verdict-red-400: #F87171;   /* secondary alert */
--verdict-red-200: #FECACA;   /* alert card border */
--verdict-red-100: #FEE2E2;   /* alert card background */
--verdict-red-50:  #FFF5F5;   /* alert rail hover */
```

Amber — Warnings & Timer (Objection / Time Pressure)

```
--verdict-amber-700: #92400E;   /* dark badge text */
--verdict-amber-600: #D97706;   /* timer warning hover */
```

```
--verdict-amber-500: #F59E0B;  /* PRIMARY WARNING ✦ */
--verdict-amber-400: #FCD34D;  /* objection badge bg */
--verdict-amber-200: #FDE68A;  /* objection card border */
--verdict-amber-100: #FEF3C7;  /* objection card background */
--verdict-amber-50:  #FFFBEB;  /* warning surface */
```

Green — Success & Active Agents

```
--verdict-green-700: #065F46;  /* dark success text */
--verdict-green-600: #059669;  /* success hover */
--verdict-green-500: #10B981;  /* PRIMARY SUCCESS ✦ */
--verdict-green-400: #34D399;  /* agent-active pulse */
--verdict-green-200: #A7F3D0;  /* confirmed flag border */
--verdict-green-100: #D1FAE5;  /* confirmed card background */
--verdict-green-50:  #ECFDF5;  /* success surface */
```

Purple — ElevenLabs / Voice AI

```
--verdict-purple-700: #4C1D95;  /* deep voice accent */
--verdict-purple-600: #7C3AED;  /* waveform hover */
--verdict-purple-500: #8B5CF6;  /* WAVEFORM FILL ✦ */
--verdict-purple-400: #A78BFA;  /* waveform mid-amplitude */
--verdict-purple-300: #C4B5FD;  /* waveform low-amplitude */
--verdict-purple-100: #EDE9FE;  /* voice badge background */
```

Neutral — Structural Grays

```
--neutral-900: #111827;
--neutral-800: #1F2937;
--neutral-700: #374151;
--neutral-600: #4B5563;
--neutral-500: #6B7280;
--neutral-400: #9CA3AF;
--neutral-300: #D1D5DB;
--neutral-200: #E5E7EB;
--neutral-100: #F3F4F6;
--neutral-50:  #F9FAFB;
--neutral-0:   #FFFFFF;
```

Tailwind Config — Full Color Registration

```typescript
// tailwind.config.ts
module.exports = {
  theme: {
    extend: {
      colors: {
        'verdict-navy': {
          900: '#0A0F1C', 800: '#0F1729', 700: '#131E30', 600: '#1E2B3C',
          500: '#253347', 400: '#2D3E54', 300: '#374B63', 200: '#4A6080',
          100: '#6B849E', 50: '#94A3B8',
        },
        'verdict-blue': {
          900: '#1E3A5F', 800: '#1D4ED8', 700: '#2563EB', 600: '#3B82F6',
          500: '#60A5FA', 400: '#93C5FD', 300: '#BFDBFE', 100: '#DBEAFE', 50: '#EFF6FF',
        },
        'verdict-red': {
          900: '#7F1D1D', 800: '#991B1B', 700: '#B91C1C', 600: '#DC2626',
          500: '#EF4444', 400: '#F87171', 200: '#FECACA', 100: '#FEE2E2', 50: '#FFF5F5',
        },
        'verdict-amber': {
```

```
        700: '#92400E', 600: '#D97706', 500: '#F59E0B', 400: '#FCD34D',
        200: '#FDE68A', 100: '#FEF3C7', 50: '#FFFBEB',
      },
      'verdict-green': {
        700: '#065F46', 600: '#059669', 500: '#10B981', 400: '#34D399',
        200: '#A7F3D0', 100: '#D1FAE5', 50: '#ECFDF5',
      },
      'verdict-purple': {
        700: '#4C1D95', 600: '#7C3AED', 500: '#8B5CF6', 400: '#A78BFA',
        300: '#C4B5FD', 100: '#EDE9FE',
      },
    },
  },
},
}
```

Color Usage Rules

Context	Token	Do	Don't
Page background	`verdict-navy-800`	All authenticated screens	Use pure black — too harsh
Panel background	`verdict-navy-600`	All cards and panels	Mix light/dark
Primary CTA	`verdict-blue-600`	ONE per screen	Use for secondary actions
Destructive action	`verdict-red-600`	End session, delete, cancel	Use for warnings — that's amber
Inconsistency alerts	`verdict-red-500`	Alert cards, borders	Swap with objection color
Objection alerts	`verdict-amber-500`	Alert cards, borders	Use for severity — amber = lesser
Confirmed flags	`verdict-green-500`	Confirmed state, agent-active	Universal success color
Score: excellent	`verdict-green-400`	Score ≥ 75	Use generically
Score: moderate	`verdict-amber-400`	Score 50–74	Use generically
Score: poor	`verdict-red-400`	Score < 50	Use generically
Voice / ElevenLabs	`verdict-purple-500`	Waveform, voice badges	Use for generic UI
Primary text	`#FFFFFF`	Headings, critical info	On dark bg only
Secondary text	`verdict-navy-50`	Labels, captions	As primary text — too low contrast
Disabled	`verdict-navy-200`	Disabled controls	Active content

Contrast Matrix (WCAG 2.1 AA Verification)

Foreground	Background	Ratio	Pass Level
`#FFFFFF` on `#0F1729`	14.7:1	✅ AAA	
`#94A3B8` on `#0F1729`	4.7:1	✅ AA	

`#6B849E` on `#0F1729`	3.6:1	✅ AA (large text)	
`#EF4444` on `#1E2B3C`	4.6:1	✅ AA	
`#F59E0B` on `#0F1729`	5.1:1	✅ AA	
`#10B981` on `#1E2B3C`	4.8:1	✅ AA	
`#8B5CF6` on `#0F1729`	4.9:1	✅ AA	
`#4A6080` on `#0F1729`	2.7:1	❌ FAIL — never use as body text	

2.2 Typography

Font Families

```
// tailwind.config.ts
fontFamily: {
  sans: ['Inter', 'system-ui', '-apple-system', 'sans-serif'],
  mono: ['JetBrains Mono', 'Fira Code', 'Consolas', 'monospace'],
},
```

```
<!-- apps/frontend/src/app/layout.tsx <head> -->
<link rel="preconnect" href="https://fonts.googleapis.com" />
<link rel="preconnect" href="https://fonts.gstatic.com" crossOrigin="anonymous" />
<link href="https://fonts.googleapis.com/css2?family=Inter:wght@300;400;500;600;700&family=Jet
```

Inter — all UI text. Optical sizing and tight letter-spacing communicate professionalism at every size from 11px captions to 48px hero text.

JetBrains Mono — exclusively for: transcript lines (speaker-tagged exchanges), confidence scores, API data, technical values. Monospace cadence makes dense legal text scannable.

Font Size Scale

```
fontSize: {
  '2xs': ['0.6875rem', { lineHeight: '1rem' }],        // 11px — timestamps, footnotes
  'xs':  ['0.75rem',   { lineHeight: '1rem' }],        // 12px — captions, metadata
  'sm':  ['0.875rem',  { lineHeight: '1.25rem' }],     // 14px — body secondary, labels, butt
  'base':['1rem',      { lineHeight: '1.5rem' }],      // 16px — body primary ✦ MAIN
  'lg':  ['1.125rem',  { lineHeight: '1.75rem' }],     // 18px — card headings
  'xl':  ['1.25rem',   { lineHeight: '1.75rem' }],     // 20px — section headings
  '2xl': ['1.5rem',    { lineHeight: '2rem' }],        // 24px — page titles
  '3xl': ['1.875rem',  { lineHeight: '2.25rem' }],     // 30px — hero text
  '4xl': ['2.25rem',   { lineHeight: '2.5rem' }],      // 36px — score number (large)
  '5xl': ['3rem',      { lineHeight: '1' }],           // 48px — score hero in brief
  '6xl': ['4rem',      { lineHeight: '1' }],           // 64px — max score display
},
```

Font Weights

```
fontWeight: {
  light:   300,  // Decorative — large display text only
  normal:  400,  // Body text, transcript lines
  medium:  500,  // UI labels, button text, tab titles
```

```
  semibold: 600,   // Card headings, alert titles, badges
  bold:      700,   // Page titles, score numbers, emphasis
},
```

Letter Spacing

```
letterSpacing: {
  tighter: '-0.05em',   // Score numbers at 5xl+
  tight:   '-0.025em',  // Page headings
  normal:  '0',         // Body text default
  wide:    '0.025em',   // All-caps labels
  wider:   '0.05em',    // Uppercase section labels
  widest:  '0.1em',     // VERDICT wordmark only
},
```

Typography Usage Table

Element	Font	Size	Weight	Color
Page title (h1)	Inter	2xl	bold	#FFFFFF
Section heading (h2)	Inter	xl	semibold	#FFFFFF
Card heading (h3)	Inter	lg	semibold	#FFFFFF
Sub-heading (h4)	Inter	base	semibold	neutral-200
Body text	Inter	base	normal	neutral-100
Body secondary	Inter	sm	normal	verdict-navy-50
Caption / metadata	Inter	xs	normal	verdict-navy-100
Timestamp	JetBrains Mono	xs	normal	verdict-navy-100
Button text (all sizes)	Inter	sm	medium	(inherits)
Badge / chip label	Inter	xs	semibold	(semantic)
Transcript — Interrogator	JetBrains Mono	sm	normal	#FFFFFF
Transcript — Witness	JetBrains Mono	sm	normal	verdict-navy-50
Score number (brief)	Inter	5xl–6xl	bold	(score color)
Confidence score	JetBrains Mono	sm	medium	verdict-navy-50
Coaching narrative	Inter	base	normal	neutral-100
Line height: relaxed	—	—	—	Only for narrative prose
Navigation item	Inter	sm	medium	verdict-navy-50
Navigation active	Inter	sm	semibold	#FFFFFF
Form label	Inter	sm	medium	neutral-200

Placeholder	Inter	base	normal	verdict-navy-200
Error message	Inter	sm	normal	verdict-red-400
Code / monospace data	JetBrains Mono	sm	normal	verdict-purple-300

2.3 Spacing Scale

```
// tailwind.config.ts
spacing: {
  px:  '1px',
  0:   '0',
  0.5: '0.125rem',  // 2px
  1:   '0.25rem',   // 4px
  1.5: '0.375rem',  // 6px
  2:   '0.5rem',    // 8px  — micro spacing
  2.5: '0.625rem',  // 10px
  3:   '0.75rem',   // 12px — compact padding
  3.5: '0.875rem',  // 14px
  4:   '1rem',      // 16px — BASE UNIT ✦
  5:   '1.25rem',   // 20px
  6:   '1.5rem',    // 24px — comfortable padding
  7:   '1.75rem',   // 28px
  8:   '2rem',      // 32px
  9:   '2.25rem',   // 36px
  10:  '2.5rem',    // 40px — button height (md)
  11:  '2.75rem',   // 44px — min touch target / button height (lg)
  12:  '3rem',      // 48px
  14:  '3.5rem',    // 56px — nav height
  16:  '4rem',      // 64px
  20:  '5rem',      // 80px
  24:  '6rem',      // 96px
  72:  '18rem',     // 288px — left panel (220px covered via custom w-[220px])
  80:  '20rem',     // 320px — right alert rail
},
```

Spacing Usage Rules

Context	Token	Example
Inline gap between icon + text	`gap-2` (8px)	Button icon + label
Compact card padding	`p-3` (12px)	Alert cards, dense rows
Standard card padding	`p-4` (16px)	Most cards
Comfortable card padding	`p-5` or `p-6`	Dashboard case cards, brief sections
Between form fields	`space-y-4` (16px)	All forms
Between section cards	`gap-6` (24px)	Page layouts
Page horizontal margin	`px-6 lg:px-8`	Standard page wrapper
Page vertical padding	`py-8`	Standard page wrapper
Section vertical separation	`mb-12` (48px)	Major page sections
Alert rail inner padding	`p-3` (12px)	Alert rail container

| Three-panel gap | `gap-0` | Panels are flush — borders provide separation |

2.4 Border Radius

```
borderRadius: {
  none:    '0',
  sm:      '0.125rem',  // 2px  — tight insets, subtle
  DEFAULT: '0.25rem',   // 4px  — input fields, chips
  md:      '0.375rem',  // 6px  — buttons ✦ BUTTON STANDARD
  lg:      '0.5rem',    // 8px  — cards, panels ✦ CARD STANDARD
  xl:      '0.75rem',   // 12px — modals, large containers
  '2xl':   '1rem',      // 16px — coaching brief hero card
  '3xl':   '1.5rem',    // 24px — large avatar frames
  full:    '9999px',    // pill — badges, chips, toggle switches
},
```

Rule table:

- `rounded-md` (6px) → all `<Button>` components
- `rounded-lg` (8px) → all `<Card>`, alert cards, panels, dropdowns
- `rounded-xl` (12px) → modals, session config overlay
- `rounded-2xl` (16px) → brief score hero, feature highlight cards
- `rounded-full` → all `<Badge>`, status dots, toggle switches, avatar circles

2.5 Shadows & Elevation

VERDICT's dark background means shadows show as border glows rather than drop shadows.

```
// tailwind.config.ts
boxShadow: {
  'card-sm':     '0 1px 2px rgba(0,0,0,0.3), 0 0 0 1px rgba(255,255,255,0.03)',
  'card':        '0 4px 6px rgba(0,0,0,0.4), 0 0 0 1px rgba(255,255,255,0.05)',
  'card-lg':     '0 10px 15px rgba(0,0,0,0.5), 0 0 0 1px rgba(255,255,255,0.07)',
  'modal':       '0 25px 50px rgba(0,0,0,0.7), 0 0 0 1px rgba(255,255,255,0.08)',
  'alert-red':   '0 0 0 1px rgba(239,68,68,0.6), 0 4px 12px rgba(239,68,68,0.2)',
  'alert-amber': '0 0 0 1px rgba(245,158,11,0.6), 0 4px 12px rgba(245,158,11,0.2)',
  'alert-green': '0 0 0 1px rgba(16,185,129,0.6), 0 4px 12px rgba(16,185,129,0.2)',
  'glow-blue':   '0 0 16px rgba(59,130,246,0.35)',
  'glow-purple': '0 0 16px rgba(139,92,246,0.35)',
  'glow-green':  '0 0 12px rgba(16,185,129,0.4)',
  'inner-focus': 'inset 0 0 0 2px rgba(59,130,246,0.7)',
},
```
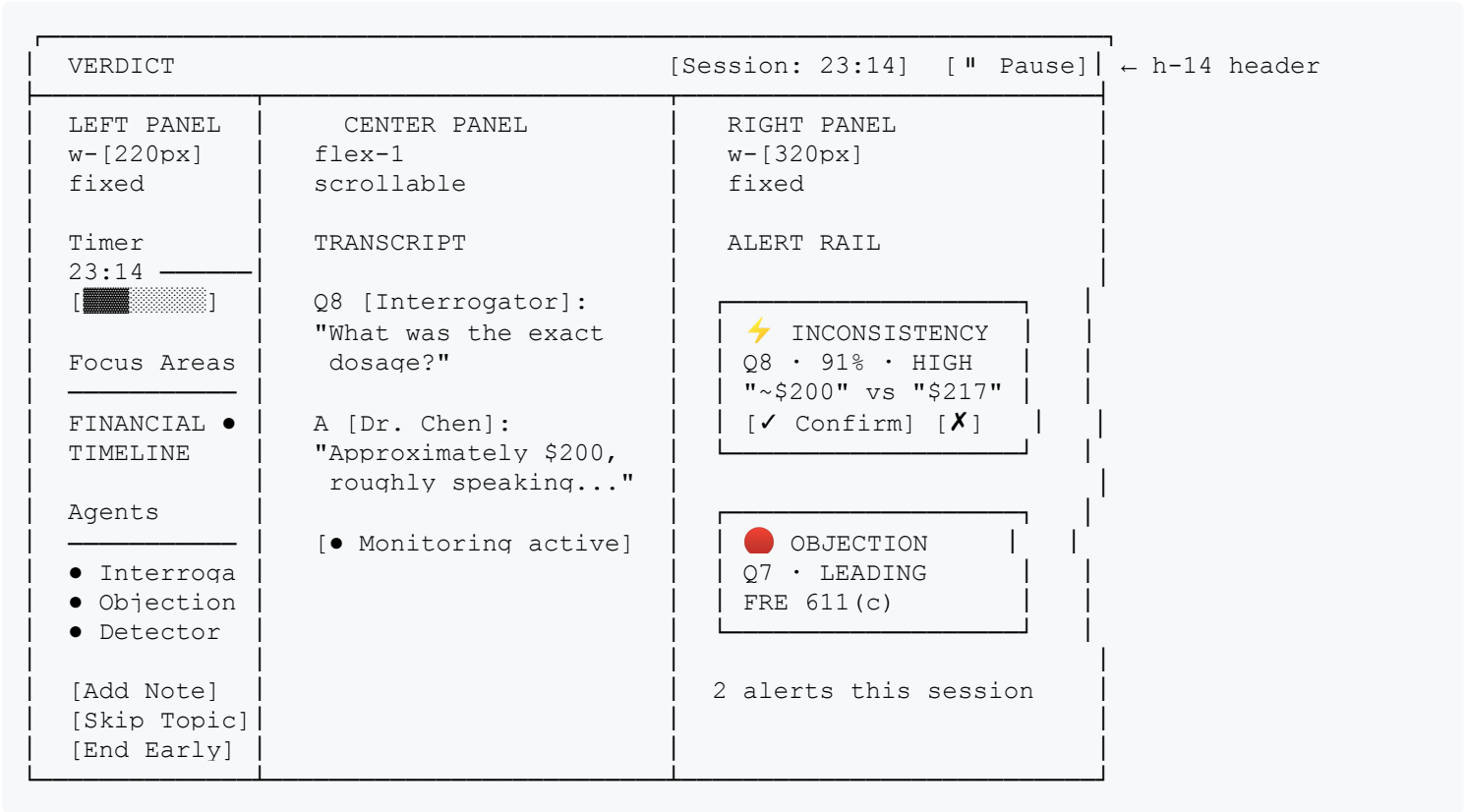
Usage:

- `shadow-card-sm` → default card resting state
- `shadow-card` → card on hover / elevated state
- `shadow-card-lg` → tooltips, floating menus
- `shadow-modal` → all modals
- `shadow-alert-red` → active inconsistency alert cards
- `shadow-alert-amber` → active objection alert cards
- `shadow-glow-purple` → ElevenLabs waveform container during playback

3. LAYOUT SYSTEM

3.1 The Three-Panel Live Session Layout

VERDICT's signature layout. Every design decision in the app builds toward making this screen functional and trustworthy.

```
┌─────────────────────────────────────────────────────────────────────┐
│  VERDICT                          [Session: 23:14]  [" Pause]│ ← h-14 header
├─────────────────────────────────────────────────────────────────────┤
│  LEFT PANEL   │   CENTER PANEL     │   RIGHT PANEL            │
│  w-[220px]    │   flex-1           │   w-[320px]             │
│  fixed        │   scrollable       │   fixed                 │
│               │                    │                         │
│  Timer        │   TRANSCRIPT       │   ALERT RAIL            │
│  23:14 ─────┐ │                    │                         │
│  [▓▓▓░░░░]   │ │   Q8 [Interrogator]:│   ┌─────────────────┐ │
│             │ │   "What was the exact│   │ ⚡ INCONSISTENCY │ │
│  Focus Areas │ │    dosage?"        │   │ Q8 · 91% · HIGH │ │
│  ─────────   │ │                    │   │ "~$200" vs "$217"│ │
│  FINANCIAL ● │ │   A [Dr. Chen]:    │   │ [✓ Confirm] [✗] │ │
│  TIMELINE    │ │   "Approximately $200,│   └─────────────────┘ │
│              │ │    roughly speaking..."│                     │
│  Agents      │ │                    │   ┌─────────────────┐ │
│  ─────────   │ │   [● Monitoring active]│   │ 🔴 OBJECTION    │ │
│              │ │                    │   │ Q7 · LEADING    │ │
│  ● Interroga │ │                    │   │ FRE 611(c)      │ │
│  ● Objection │ │                    │   └─────────────────┘ │
│  ● Detector  │ │                    │                         │
│              │ │                    │   2 alerts this session │
│  [Add Note]  │ │                    │                         │
│  [Skip Topic]│ │                    │                         │
│  [End Early] │ │                    │                         │
└─────────────────────────────────────────────────────────────────────┘
```

// apps/frontend/src/app/cases/[caseId]/session/[sessionId]/live/page.tsx export default function LiveSessionPage() { return (
{/* Fixed header — h-14 (56px) */}
VERDICT

  `}>Pause`   `End Session`

{/* Three-panel body — fills remaining viewport height */}
{/* LEFT — Session controls, fixed 220px */}
{/* CENTER — Live transcript, flexible width */}
{/* RIGHT — Alert rail, fixed 320px */}
); }

3.2 Standard Authenticated Page Layout

// apps/frontend/src/components/layout/PageLayout.tsx export function PageLayout({ title, description, actions, children, maxWidth = '7xl', }: { title: string; description?: string; actions?: React.ReactNode; children: React.ReactNode; maxWidth?: '5xl' | '6xl' | '7xl' | 'full'; }) { return (
{/* Page header */}

{title}

{description && (

{description}

)}
{actions && (
{actions}

```
)}
{/* Main content */}
{children}
); }
```

3.3 Responsive Breakpoints

```
// tailwind.config.ts
screens: {
  'sm':  '640px',    // Mobile landscape
  'md':  '768px',    // Tablet
  'lg':  '1024px',   // Desktop minimum — required for live session
  'xl':  '1280px',   // Standard desktop ✦ PRIMARY TARGET
  '2xl': '1536px',   // Wide desktop / dual-monitor
},
```

3.4 Common Layout Patterns

// 1. Case cards dashboard grid
{cases.map(c =>)}
// 2. Two-column: main content + sidebar
{/* Primary content */}
{/* Sidebar — stick on scroll */}
// 3. Case detail tabs with content area
{/* Tab content */}
// 4. Centered auth layout (login, invite accept)
// 5. Coaching brief layout (single-column, constrained)
// 6. Witness view — clean, minimal (no alert data)

3.5 Global Navigation

// apps/frontend/src/components/layout/GlobalNav.tsx export function GlobalNav() { const pathname = usePathname(); return (
{/* Skip link — keyboard users */} Skip to main content {/* Wordmark */} VERDICT {/* Divider */}
{/* Nav links */}
} /> } />
{/* Right — firm badge + user menu */}
); } function NavLink({ href, label, icon }: { href: string; label: string; icon: React.ReactNode }) { const pathname = usePathname(); const isActive = pathname === href || pathname.startsWith(href + '/'); return ({icon} {label}); }

---

4. COMPONENT LIBRARY

4.1 Button

// apps/frontend/src/components/ui/Button.tsx import { cva, type VariantProps } from 'class-variance-authority'; import { Loader2 } from 'lucide-react'; import { cn } from '@/lib/utils'; const buttonVariants = cva(['inline-flex items-center justify-center gap-2', 'font-medium rounded-md select-none', 'transition-all duration-150 ease-out', 'focus-visible:outline-none focus-visible:ring-2 focus-visible:ring-offset-2 focus-visible:ring-offset-verdict-navy-800', 'disabled:opacity-40 disabled:cursor-not-allowed disabled:pointer-events-none',], { variants: { variant: { primary: ['bg-verdict-blue-600 text-white', 'hover:bg-verdict-blue-700 active:bg-verdict-blue-800', 'focus-visible:ring-verdict-blue-500', 'shadow-card-sm hover:shadow-card',], secondary: ['bg-verdict-navy-600 text-white border border-verdict-navy-300/50', 'hover:bg-verdict-navy-500 hover:border-verdict-navy-300/80', 'focus-visible:ring-verdict-navy-300',], outline: ['bg-transparent text-verdict-blue-400 border border-verdict-blue-600/50', 'hover:bg-verdict-blue-600/10 hover:border-verdict-blue-500 hover:text-verdict-blue-300', 'focus-visible:ring-verdict-blue-500',], ghost: ['bg-transparent text-verdict-navy-50', 'hover:bg-verdict-navy-600 hover:text-white', 'focus-visible:ring-verdict-navy-300',], danger: ['bg-verdict-red-600 text-white', 'hover:bg-verdict-red-700

active:bg-verdict-red-800', 'focus-visible:ring-verdict-red-500', 'shadow-card-sm',], 'danger-outline': ['bg-transparent text-verdict-red-400 border border-verdict-red-600/50', 'hover:bg-verdict-red-600/10 hover:border-verdict-red-500', 'focus-visible:ring-verdict-red-500',], success: ['bg-verdict-green-600 text-white', 'hover:bg-verdict-green-700 active:bg-verdict-green-800', 'focus-visible:ring-verdict-green-500',], }, size: { xs: 'h-7 px-2.5 text-xs', sm: 'h-8 px-3 text-sm', md: 'h-10 px-4 text-sm', // DEFAULT ✦ lg: 'h-11 px-6 text-base', // Primary CTAs xl: 'h-12 px-8 text-base', // Hero CTAs icon: 'h-10 w-10', 'icon-sm': 'h-8 w-8', }, }, defaultVariants: { variant: 'primary', size: 'md' }, }); interface ButtonProps extends React.ButtonHTMLAttributes, VariantProps { loading?: boolean; icon?: React.ReactNode; iconPosition?: 'left' | 'right'; } export function Button({ className, variant, size, loading, disabled, icon, iconPosition = 'left', children, ...props }: ButtonProps) { return (

{loading && } {!loading && icon && iconPosition === 'left' && ({icon})} {children} {!loading && icon && iconPosition === 'right' && ({icon})}

); } // When to use each variant: // primary → ONE per screen. The most important action. [Start Session], [Generate Brief] // secondary → Supporting actions alongside primary. [View Details], [Download] // outline → Lower-emphasis alternatives. [Cancel], tab-like actions // ghost → Navigation, icon buttons, inline actions in dense UI // danger → Irreversible destructive actions: [End Session], [Delete Case] // danger-outline → Softer destructive: "confirm before proceeding" dialogs // success → Confirm alert, [Mark Complete], [Accept]

4.2 Input Fields

// apps/frontend/src/components/ui/Input.tsx import { Eye, EyeOff, AlertCircle, CheckCircle2 } from 'lucide-react'; interface InputProps extends React.InputHTMLAttributes { label?: string; error?: string; success?: string; helperText?: string; inputSize?: 'sm' | 'md' | 'lg'; } const inputBaseClasses = ['w-full bg-verdict-navy-600 border rounded', 'text-white placeholder:text-verdict-navy-200', 'transition-all duration-150', 'focus:outline-none focus:ring-2 focus:ring-offset-0', 'disabled:opacity-40 disabled:cursor-not-allowed',].join(' '); const inputSizeClasses = { sm: 'h-8 px-3 text-sm', md: 'h-10 px-3 text-sm', // DEFAULT ✦ lg: 'h-12 px-4 text-base', }; export function Input({ id, label, error, success, helperText, required, inputSize = 'md', className, type, ...props }: InputProps) { const [showPw, setShowPw] = React.useState(false); const inputId = id ?? React.useId(); const errorId = `${inputId}-error`; const helperId = `${inputId}-helper`; const state = error ? 'error' : success ? 'success' : 'default'; const actualType = type === 'password' ? (showPw ? 'text' : 'password') : type; const stateClasses = { default: 'border-verdict-navy-300/50 hover:border-verdict-navy-300 focus:ring-verdict-blue-500 focus:border-verdict-blue-500', error: 'border-verdict-red-500 focus:ring-verdict-red-500 focus:border-verdict-red-500', success: 'border-verdict-green-500 focus:ring-verdict-green-500 focus:border-verdict-green-500', }; const needsIcon = state !== 'default' && type !== 'password'; return (
{label && ({label} {required && *} {required && (required)})}

{/* Password toggle */} {type === 'password' && (

setShowPw(p => !p)} aria-label={showPw ? 'Hide password' : 'Show password'} > {showPw ? : })} {/* State icons */} {state === 'error' && type !== 'password' && ( )} {state === 'success' && type !== 'password' && ( )}
{error && (

{error}

)} {success && !error && (

{success}

)} {helperText && !error && !success && (

{helperText}

)}
); } // Textarea variant export function Textarea({ id, label, error, helperText, required, rows = 4, className, ...props }: React.TextareaHTMLAttributes & { label?: string; error?: string; helperText?: string; required?: boolean; }) { const inputId = id ?? React.useId(); const errorId = `${inputId}-error`; return (

{error && (
<p id=
{label && ({label} {required && *})}