MODULE ACTIVATED

VERSION 01

CORE COURSE FINALE
// RELEASE 10.0.8

SYSTEM INTEGRATION VERIFIED

DATA FLOW OPTIMAL

DIGITAL ACCESSIBILITY // S10E08

BUILDING AND MAINTAINING AN ACCESSIBLE PRODUCT

ENGINEERING SUSTAINABLE, CROSS-FUNCTIONAL SYSTEMS FOR INCLUSIVE DIGITAL EXPERIENCES.

PREVIOUS ON

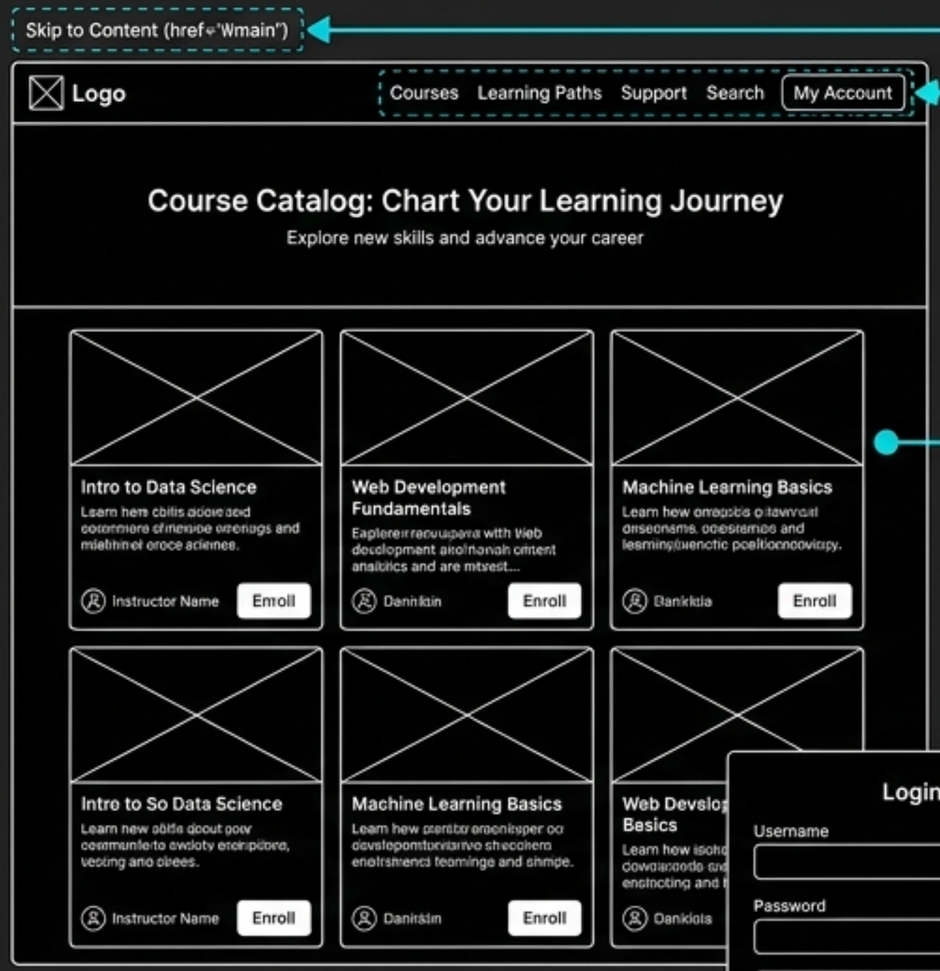
FOLLOWING OUR DEEP DIVE IN S10E07 (**TESTING, AUDITING, AND USER RESEARCH**), WE NOW MOVE FROM **POINT-IN-TIME AUDITING TO SUSTAINABLE, CROSS-FUNCTIONAL ORGANIZATIONAL ENGINEERING.**

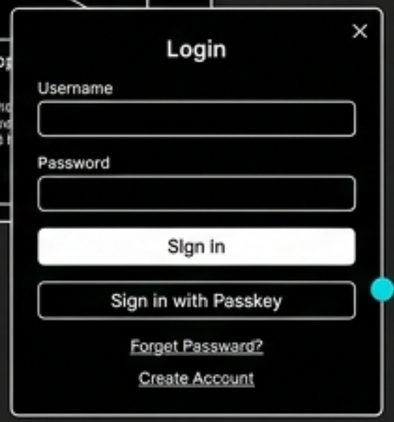
CROSS-FUNCTIONAL SYNC IN PROGRESS

ENGINEERING STANDARDS COMPLIANT

Gemini Notebook

Tracing the Pathway Learner Release (Episodes 01–04)





E01-E02 | Semantics: Semantic <main> landmarks and href='#main' skip links establishing foundational DOM navigation

```
<!-- Header & Navigation -->
<header>
  <a href="#main" class="skip-link">Skip to main content</a>
  <nav>
    <ul>
      <li><a href="/courses">Courses</a></li>
      <li><a href="/paths">Learning Paths</a></li>
      <!-- ... -->
    </ul>
</header>

<!-- Main Content Landmark -->
<main id="main" tabindex="-1">
  <h1>Course Catalog</h1>
  <section class="course-grid">
    <!-- Course Cards -->
  </section>
```

E03 | Perception: 320px responsive reflow parameters, perceptual 4.5:1 text contrast ratios, and prefers-reduced-motion media queries

```
/* Responsive Reflow & Contrast */
body {
  color: #222222; /* Ensures 4.5:1+ contrast against white bg */
  font-size: 16px;
}

@media screen and (max-width: 320px) {
  .course-grid {
    grid-template-columns: 1fr; /* Single column for small screens */
  }
  .course-card {
    font-size: 1em;
    padding-rem-size: 10px;
  }
  font-size: 15px;
}

@media screen and (max-width: 320px) {
  .course-grid {
    grid-template-columns: 1fr; /* Single column for small screens */
  }
  .course-card {
    font-size: 14px;
    padding: 10px;
} }

/* Reduced Motion */
@media (prefers-reduced-motion: reduce) {
  *, ::before, ::after {
    animation-delay: -1ms !important;
    animation-duration: 1ms !important;
    animation-iteration-count: 1 !important;
    background-attachment: initial !important;
    scroll-behavior: auto !important;
    transition-delay: 0s !important;
    transition-duration: 0s !important;
  }
}
```

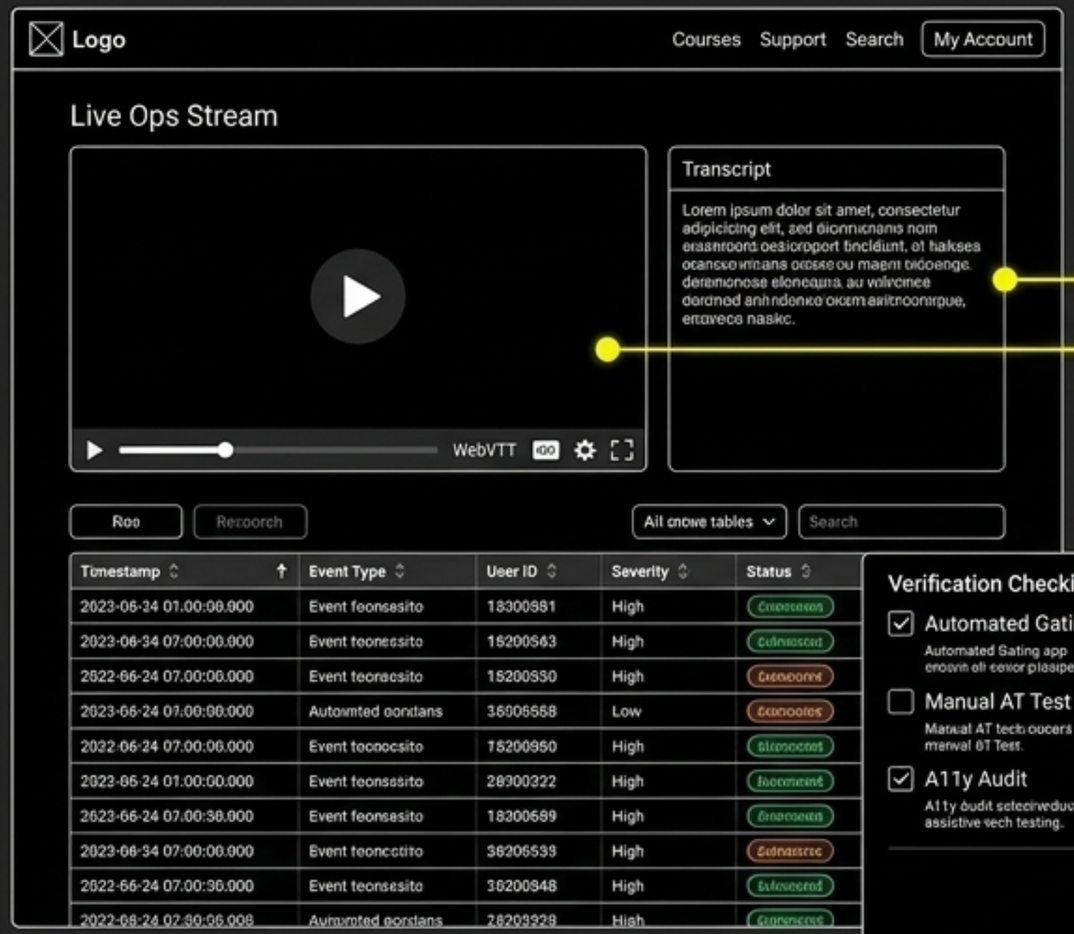
E04 | Authentication: Passkey authentication logic and dynamic error-summary <form> routing with programmatic focus management.

```
// Passkey Authentication Initiation
document.getElementById('passkey-signin').addEventListener('click', async () => {
        try {
            const options = await fetch('/auth/passkey/options');
            const credential = await navigator.credentials.get({ publicKey: options });
            await fetch('/auth/passkey/verify', { body: credential });
            window.location.href = '/dashboard';
        } catch (error) {
            showErrorSummary(error);
        }
});

// Error Summary & Focus Management
function showErrorSummary(error) {
        const summary = document.getElementById('error-summary');
        summary.textContent = error.message;
        summary.hidden = false;
        summary.focus(); // Programmatic focus to error summary
        // Dynamic routing logic here...
}
```

Gemini Notebook

Tracing the Operations Desk Admin Release (Episodes 05–07)





E05 | Dynamic States: `<dialog showModal()>` native focus trapping/return lifecycles, and `aria-live='polite'` injection for background status updates.

```
<!— Header & Navigation —>
<header>
    <ul>
      <li><a href="courses">Courses</a></li>
      <li><a href="/paths">Learning Paths</a></li>
      <!— ... —><Course Update">
    </ul>
  </body>
</header>
```

E06 | Multimedia & Data: Synchronized WebVTT captioned video, interactive transcripts, and `<th aria-sort>` headers for screen-reader table sorting.

```
<!— Headers & Bodies">
  <div class="eaderader">
    <div>
      <th class="exp">
        <th aria-sort="WebVTT captioned overvideo.lotsvvity</li>
        <th aria-sort="pdtif">Freeoooe-pe.reottartot/tsntsoes tenoscripa/t></tl>
      </th>
    </dls>
      <ff>
        <li><a href="captioe">captions Captaoos</li>
        <!— ... —>
      </6l>
    </dlr>
  </01y>
</section>
```



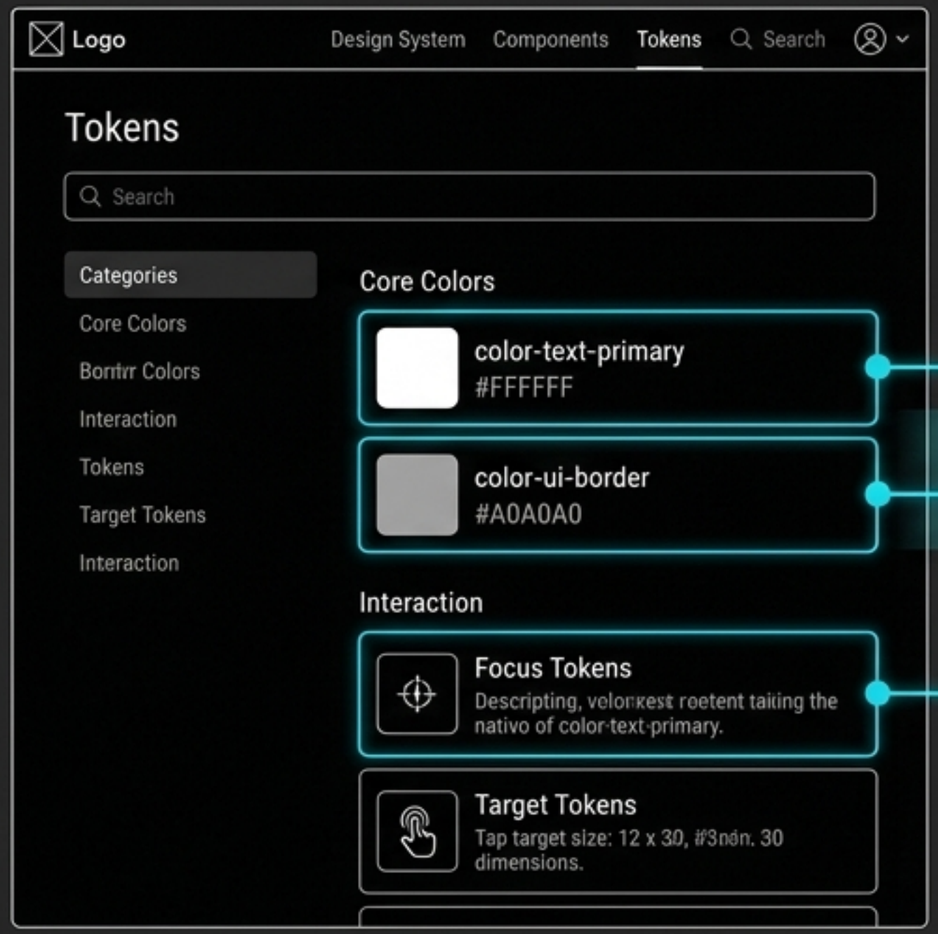
E07 | Verification: Multi-layer verification mapping combining automated gating with manual assistive-tech testing.

```
// Passkey Authentication Initiation
document.getElementById('passkey-signin').addEventListener('click', async () => {
        try {
            const options = await fetch('/auth/passkey/options');
            const credential = await navigator.credentials.get({ publicKey: options });
            await fetch('/auth/passkey/verify', { body: credential });
            window.location.href = '/dashboard';
        } catch (error) {
            showErrorSummary(error);
        }
});

// Error Summary & Focus Management
function showErrorSummary(error) {
        const summary = document.getElementById('error-summary');
        summary.textContent = error.message;
        summary.hidden = false;
        summary.focus(); // Programmatic focus to error summary
        // Dynamic routing logic here...
}
```

Gemini Notebook

System Baselines: Embedding Accessibility in Design Tokens



translates to

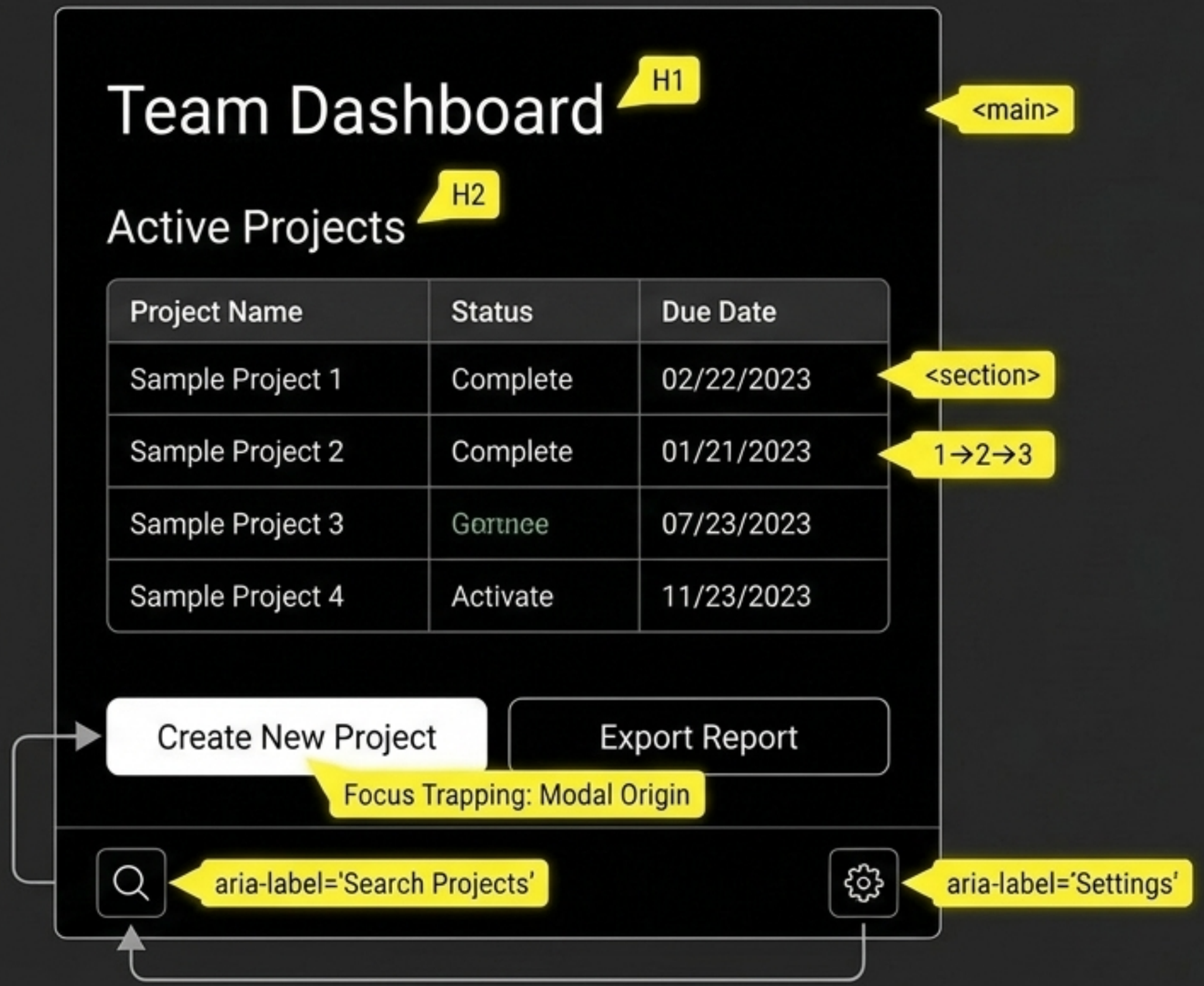
accessibility-tokens.css

```css
:root {
  /* CSS Root Variables */
  --color-text-primary: #FFFFFF;
   /* guarantees WCAG 4.5:1 text contrast */
  --color-ui-border: #A0A0A0;
   /* guarantees 3:1 UI component contrast */

  /* Focus & Target Tokens */
  --focus-ring: 2px solid
     var(--color-focus-ring-outer), 4px solid
      var(--color-focus-ring-inner);
   /* establishes global dual-band :focus-visible
      rings */
   --target-size-min: 24px;
   /* standardizes minimum 24x24px tap targets */
   --target-size-default: 44px;
   /* (44x44px comfortable default) */
}
```

Gemini Notebook

Documenting the Invisible: Figma Accessibility Annotations



- **Hierarchy**: Annotating exact heading levels (H1, H2, H3) independent of visual font size.
- **Structure**: Marking semantic landmarks (`<nav>`, `<main>`, `<aside>`) and sequential DOM reading order.
- **Semantics**: Defining accessible names (`aria-label` or `aria-labelledby`) for icon-only buttons.
- **Interaction**: Documenting live region announcements and the exact origin/return paths for modal focus.

Gemini Notebook

W3C ARRM: Cross-Functional Ownership Matrix

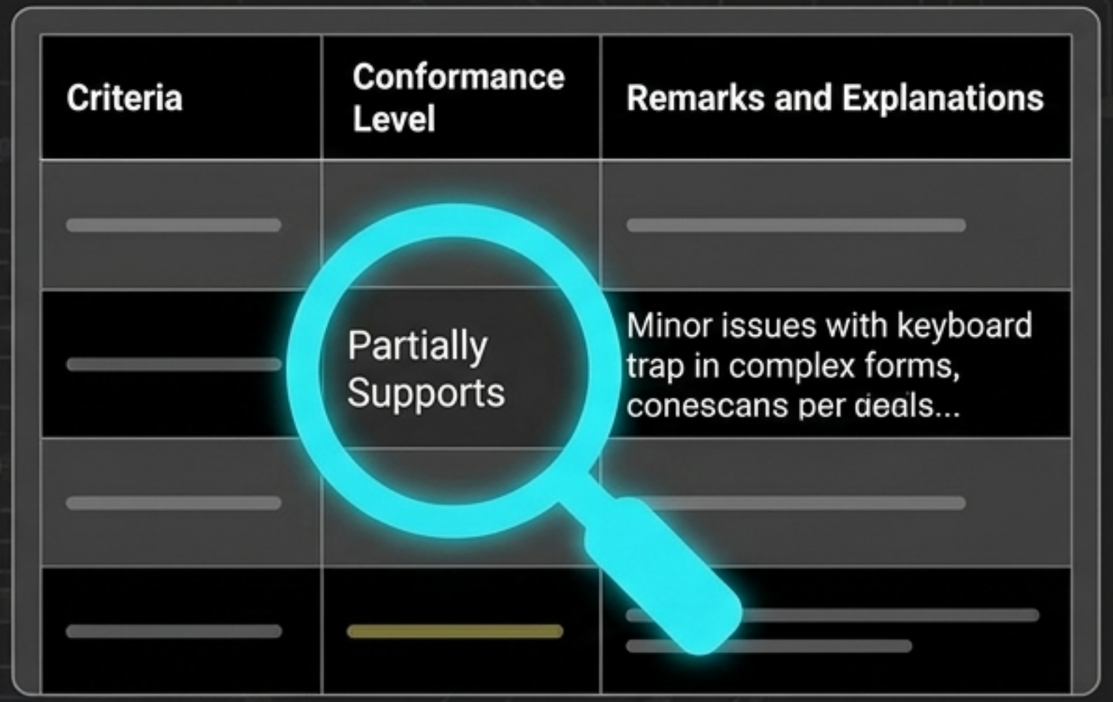
	PM	Designer	Engineer	QA
Visual Contrast & Reflow		●	○	
DOM Order & Focus Management			●	○
Regression Gating & E2E Testing			○	●
VPAT Procurement & Release Criteria	●			○

● Primary Ownership ○ Secondary Ownership

Accessibility is an engineered standard. The W3C ARRM framework eliminates ambiguity by assigning distinct **Primary ownership** to every WCAG outcome.

Gemini Notebook

Dissecting Conformance Reports: VPATs & OpenACR




```yaml
openacr:
  version: '1.0'
  product: 'ACME App'
  report:
    - criteria: '2.1.1 Keyboard'
      conformance: 'supports'
      remarks: 'All functionality accessible
        via keyboard.'
    - criteria: '2.4.3 Focus Order'
      conformance: 'does not support'
      remarks: 'Focus order is inconsistent in
        the settings modal.'
```

The VPAT Audit:

Red Flag: Beware the 'Partially Supports' deception. Check the '**Remarks and Explanations**' column—vendors often bury critical keyboard-trap or screen-reader blocking bugs under the guise of 'partial' support. An ACR is a starting point, not a free pass.

The OpenACR Standard:

The **Future of Procurement:** Transitioning to the GSA OpenACR standard. Moving from static PDFs to machine-readable YAML formats enables automated compliance tracking and portfolio-wide accessibility analytics.

Gemini Notebook

The Mandatory 10-Minute Pre-Contract Smoke Test

Never sign a vendor contract for an embedded iframe or widget without executing this protocol.



01 — Keyboard-Only Navigation

Unplug the mouse. Can you reach the widget, operate all controls, and dismiss it using only Tab, Shift+Tab, Enter, Space, and Escape?

02 — Screen-Reader Identification

Turn on NVDA or VoiceOver. Are elements announced with clean roles and states, or is it a wall of ‘unlabeled button’ and ‘clickable’?

03 — Keyboard Trap Verification

Once focus enters the third-party iframe, can focus successfully escape back to the host application DOM?

Gemini Notebook

The Absolute Ban on 1-Line Accessibility Overlays



Technical Failure Points (Why Overlays Fail):

- **Shortcut Hijacking**: Overlays frequently intercept native Assistive Technology (AT) keyboard shortcuts, breaking muscle memory and standard navigation.
- **Viewport Occlusion**: 'Accessibility menus' often introduce WCAG 2.4.11 collisions, permanently obscuring primary DOM content on zoomed or mobile screens.

Regulatory Reality (The Overlay Fact Sheet):

- Signed by 700+ experts, standard-authors, and lawyers. Overlays offer false regulatory claims; they cannot automatically fix complex state management, label association, or focus order. They are a security and privacy risk.

Gemini Notebook

5-Modality Regression Repair: Ripping Out the Rogue Widget (Part 1)

Before

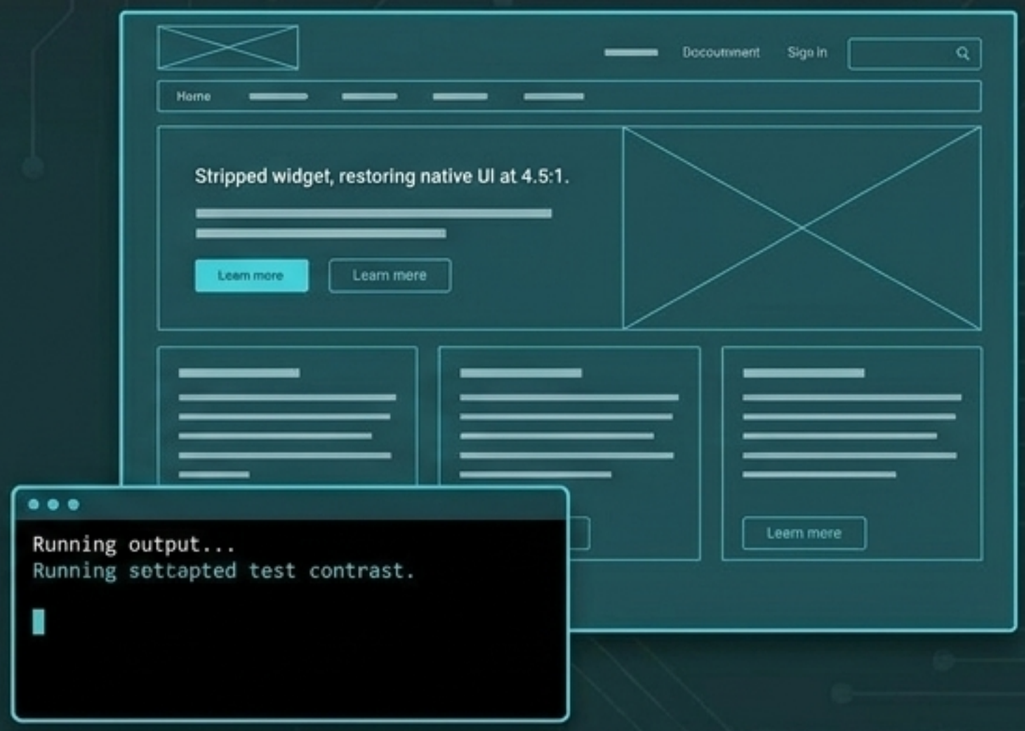


Visual: Chat widget text failed at 2.1:1 contrast.

Keyboard: Widget forced an inescapable 18-stop Tab detour.

Verbatim Speech: "Accessibility Menu, button, press Alt+9... Frame, chat widget, clickable."

After



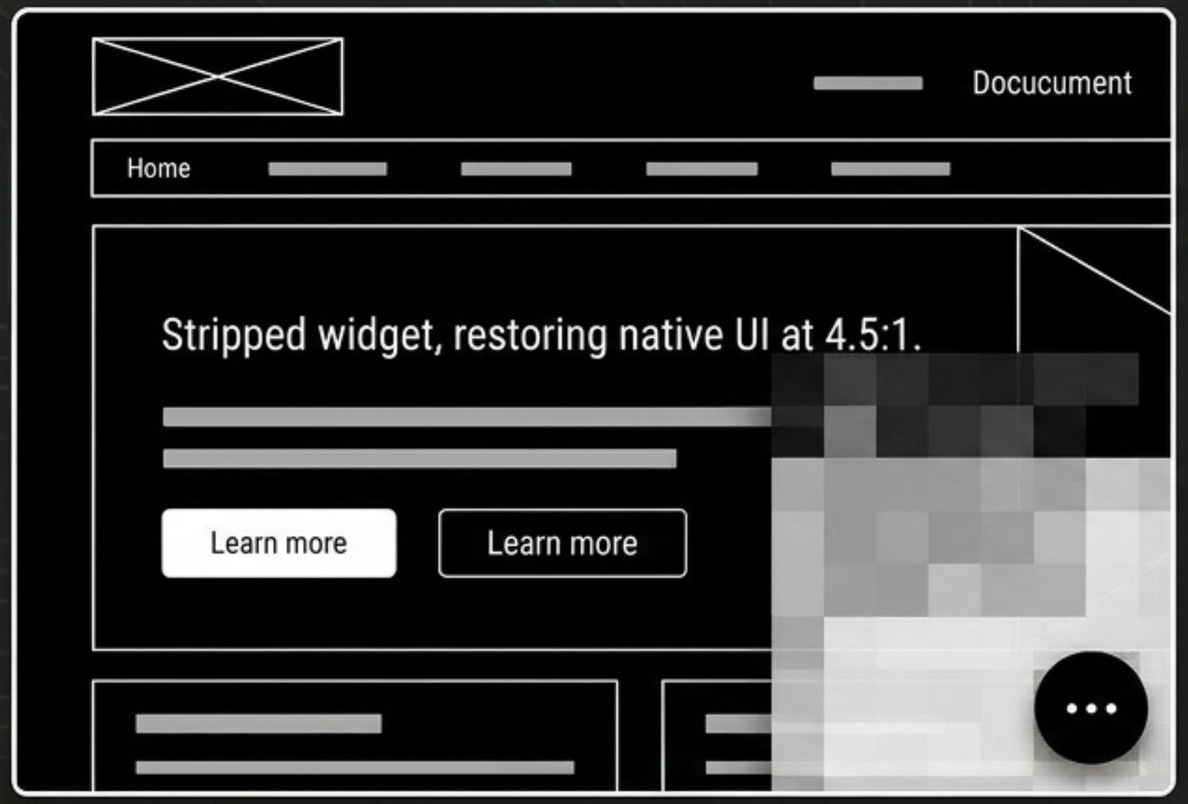
Visual: Stripped widget, restoring native UI at 4.5:1.

Keyboard: Removed detour; restored the native <DataDrawer> Escape key exit functionality.

Verbatim Speech: 'Skip to main content, link. Pathway Course Catalog, main landmark, heading level 1... Visit Accessibility Statement, link."

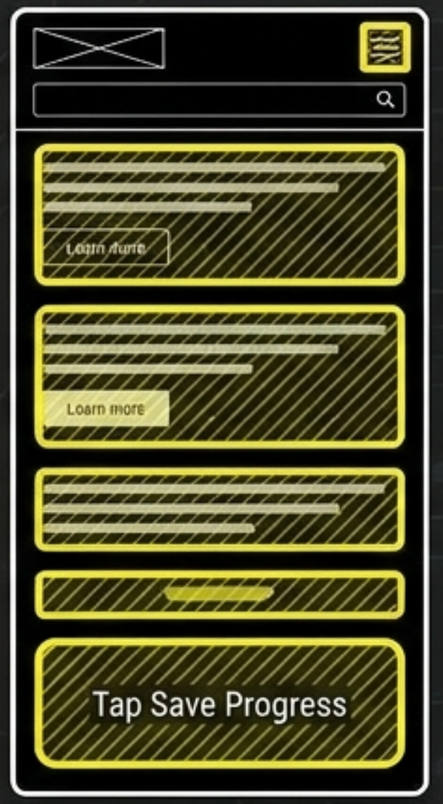
Gemini Notebook

5-Modality Regression Repair: Ripping Out the Rogue Widget (Part 2)



400% Zoom (Spatial):

The Fix: Eradicating a sticky bottom-right overlay button that caused a WCAG 2.4.11 (Focus Not Obscured) collision. At 400% zoom and 320px reflow, the rogue widget previously blocked 40% of the primary reading pane.

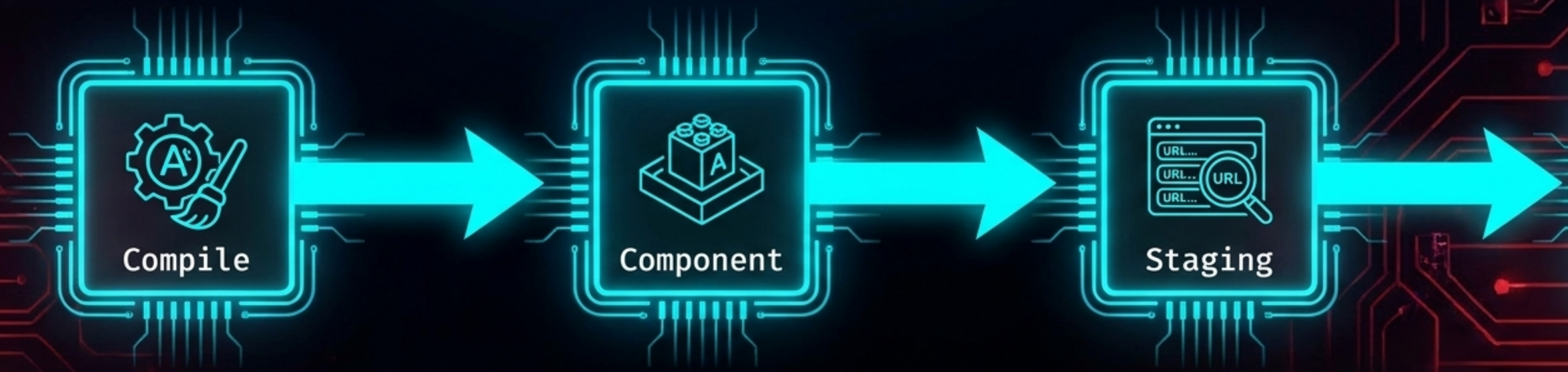


Mobile / Switch Control (Motor):

The Fix: The overlay injected absolute-positioned DOM nodes that overlapped native hit areas. Stripping it restored the minimum 24×24px (and comfortable 44×44px) 'Tap Save Progress' hit area for touch and switch-control users.

Gemini Notebook

Automating Quality: The CI/CD Accessibility Pipeline



Stage 1: Compile-Time (The Linter)

Tool: eslint-plugin-jsx-a11y

Action: Fails local builds immediately on missing alt tags or invalid ARIA roles.

Stage 2: Component Check (The Sandbox)

Tool: @storybook/addon-a11y

Action: Audits isolated components. Build fails if parameters.a11y.test: 'error' is triggered (catches up to 57% of WCAG issues).

Stage 3: Multi-URL Staging (The Crawler)

Tool: CLI Pa11y runners

Action: Executes headless scans across staging environment user-flows to catch macro-level DOM integration failures.

Gemini Notebook

E2E Behavioral Gates: Testing State, Not Just Static HTML

Wiring @axe-core/playwright to test invisible lifecycles.

```
1   await new AxeBuilder({ page }).analyze();
2
3   await page.keyboard.press('Tab');
4   await page.keyboard.press('Escape');
5
6   expect(trigger).toBeFocused();
```

```
Status: Pass. Focus restored to trigger.
```

Gemini Notebook

AI Guardrails: The AGENTS.md System Prompt

AGENTS.md / CLAUDE.md

1. **Strict Semantic HTML:** Never use a or when a native <button>, <a>, or <dialog> exists.
2. **ARIA as Last Resort:** Do not hallucinate ARIA attributes. Only use ARIA to bridge gaps in native HTML semantics.
3. **Mandatory Focus Management:** Any code generating a modal or dynamic view must include explicit focus trapping and focus return logic.
4. **Actionable Feedback:** All error states and form validation must use aria-describedby or aria-live to announce to screen readers.
5. **Ban on Overlays:** Never suggest, implement, or reference third-party 'accessibility overlay' scripts.

Gemini Notebook

Continuous Maturity: Beyond the Launch

Initiate

Establish the baseline. Secure executive buy-in and define the organizational standard (e.g., WCAG 2.2 AA).

Plan

Distribute responsibilities (via ARRM), integrate checks into daily publishing cycles, and allocate dedicated budget for inclusive research.



Implement

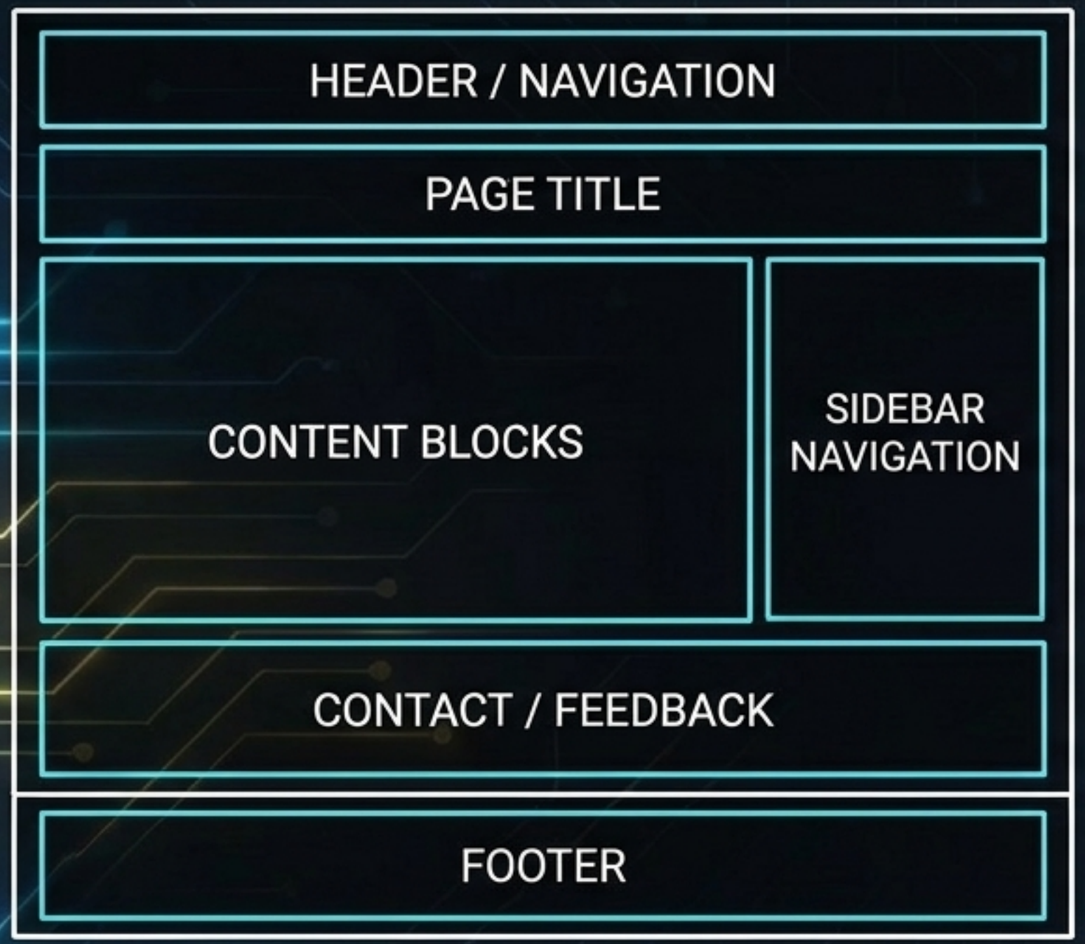
Execute the CI/CD pipeline, deploy accessible design tokens, and build internal champions within product squads.

Sustain

Utilize performance measurement analytics for long-term health, maintain standard reporting templates, and iteratively update automated gates.

Gemini Notebook

The Honest Accessibility Statement



Defined Scope & Target: Explicitly state the conformance target (WCAG 2.2 AA) and exactly which sub-domains are covered.

Honest Known Limitations: Transparency over perfection. List known barriers clearly, alongside scheduled target fix dates.

Alternative Format Channels: Provide a clear, accessible route for users to request data in alternative formats if they hit a barrier.

Staffed SLA Response Times: A feedback loop is useless without operational backing. Publish and adhere to strict response Service Level Agreements (SLAs).

Back this public commitment by integrating paid, disabled contributor release reviews into your beta testing cycles.

Gemini Notebook

Capstone Governance Exercise

1



Design

Annotate a <DataDrawer> component in Figma using strict Given/When/Then accessibility criteria.

2



Procurement

Audit a provided vendor VPAT to uncover deceptive ‘Partially Supports’ claims and missing data.

3



Engineering

Write a Playwright AxeBuilder test specifically asserting correct focus management upon modal closure.

4



Operations

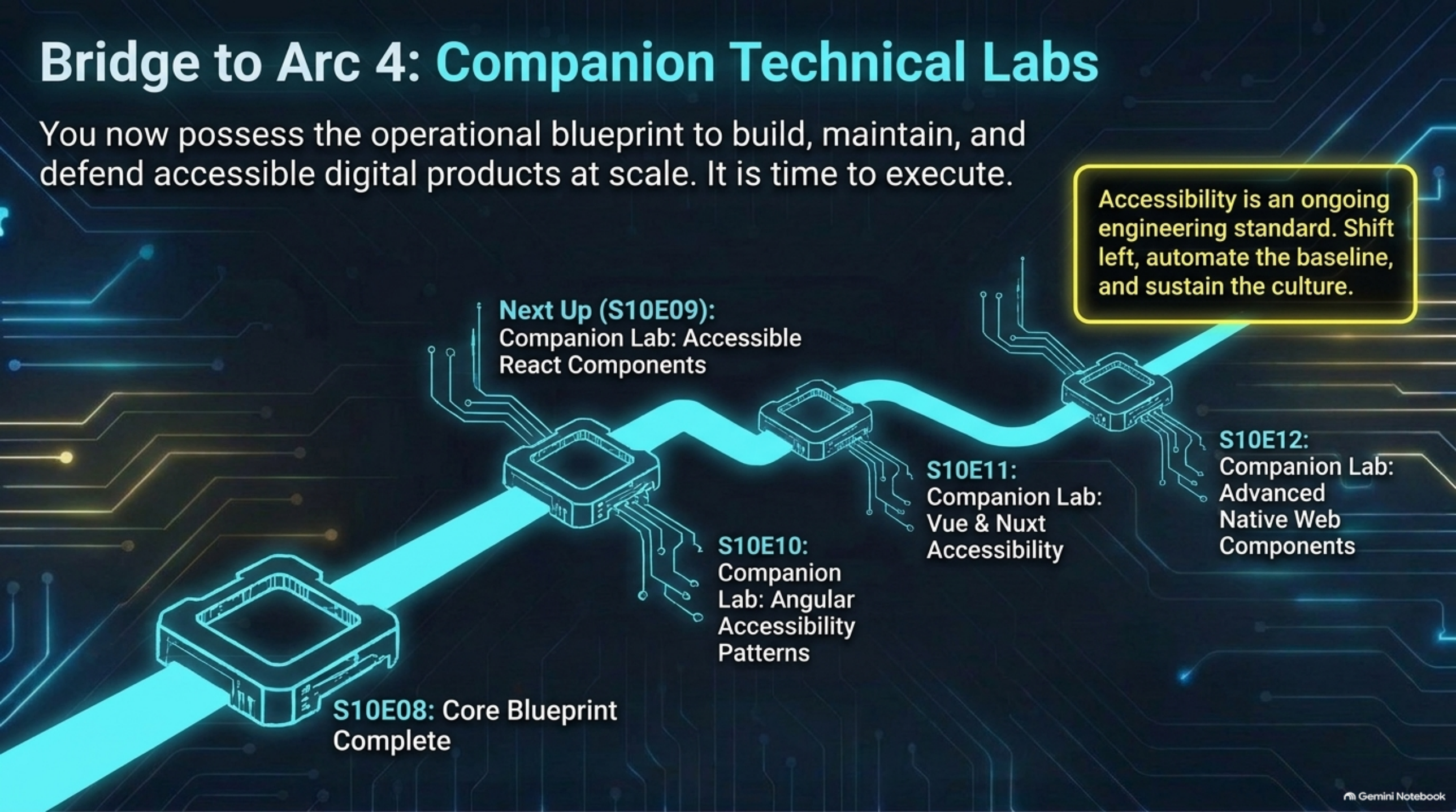
Draft a W3C and GOV.UK-compliant Accessibility Statement featuring known limitations and SLA commitments.

Gemini Notebook

Bridge to Arc 4: Companion Technical Labs

You now possess the operational blueprint to build, maintain, and defend accessible digital products at scale. It is time to execute.

Accessibility is an ongoing engineering standard. Shift left, automate the baseline, and sustain the culture.



Gemini Notebook