

IRIS Handoff Nov 28 2025

IRIS Website Generation – Handover Summary (Week 1)

Context

We've been setting up a repeatable way for IRIS to generate inbound-focused web pages, so we can move FVG from outbound to inbound. Since you already know the overall goal; this handover is about the concrete assets, workflow, and direction so you can take it forward.

The core idea is: **IRIS doesn't just get a prompt - it gets context + examples.**

We define standards in text "context" files, back them with real code examples (components and full pages), and then ask IRIS to assemble new pages by following those patterns. Fine-tuning happens by updating the context, not retraining the model.

1. Methodology

a) Context-driven generation

- We use **context files** to encode:
 - IRIS block requirements (hero, pain, solution, role boundaries/scope, security, timeline, growth track, SEO, etc.).
 - FVG defaults (logo, phone, email, booking link).
 - Domain rules (e.g. legal/regulatory constraints).
- These contexts can be **branched per domain** (legal, RevOps, etc.) so each domain keeps its own standards while using the same engine.

b) Components + examples

- We don't rely on GPT to "invent" layout.
- Instead we:
 - Provide **sample pages** that show the full structure.
 - Provide a **component library** (HTML + CSS) that defines how sections should look and behave.
- IRIS then fills in content and wiring, using the components and IRIS rules as a contract.

c) Multi-LLM friendly

- GPT is used where it's strong (content, structure, IRIS block logic).
 - Design/visual polish can be done with other models/tools.
 - Because everything is expressed as **HTML, CSS and placeholders**, we can mix and match LLMs without breaking the pipeline.
-

2. Current Assets

1) IRIS legal instructions

- File: `instructions legal.txt`
- What it does:
 - Defines the full IRIS landing-page structure for legal roles:
 - Hero, Our Customers, Ratings/Reviews, Pain, Solution, Role Boundaries, Role Scope, Security, Timeline, Growth Track, Cost (if numbers exist), SEO.
 - Enforces:
 - Above-the-fold rule (Hero + Customers/Reviews/Pain).
 - Dual CTA layout (primary + secondary) and colour rules.
 - Use of visible `[Insert ...]` placeholders instead of inventing data.
 - FVG defaults for logo, phone, email, calendar link.
- This is the **master context** IRIS reads whenever we build a legal/role page.

2) Legal example page (already built)

- File: `fvg-from-gpt/paralegal01-sonnet.php`
- What it is:
 - A complete WordPress PHP template for **“Legal Operations Associate - Virtual Paralegal Services”**.
 - Starts with the correct `// Template Name` comment.
 - Implements all IRIS blocks from the legal instructions:
 - Outcome-driven hero with dual CTAs and SEO meta.
 - Pain, solution, role boundaries (Included / Stretch / Excluded), role scope.
 - Security model, onboarding timeline, growth track, SEO keywords.
 - Includes its own layout/CSS aligned to the IRIS design rules.
- How to use it:
 - As a **canonical working example** when we ask IRIS to create new legal pages.
 - As a pattern for other role pages (e.g., litigation VA, intake VA).

3) Generic landing-page components

- Directory: `components/`
- What's inside:
 - **CSS**

- `css/global.css`: reset, variables, typography, layout utilities, key animations.
- `css/components.css`: header, hero, services, testimonials, FAQ, final CTA, footer, sticky CTAs, etc.
- **HTML partials** (`partials/`):
 - `header.html`, `notification-bar.html`
 - `hero.html`
 - `trust-section.html` (client logos + reviews widget)
 - `services.html` / `service-card.html`
 - `why-choose-us.html` / `feature-item.html`
 - `process-section.html` / `process-step.html`
 - `testimonials.html` / `testimonial-item.html`
 - `faq.html` / `faq-item.html`
 - `final-cta.html`
 - `footer.html`
- **Docs & samples**
 - `README.md`: documents every placeholder (e.g. `{{HERO_TITLE_LINE_1}}`, `{{STEP_NUMBER}}`) and how to use the components.
 - `sample-page.html`: a full page wiring all partials together in the exact order we want (notification bar → header → hero → trust → services → why us → testimonials → FAQ → final CTA → footer).
 - `all-components-sample.html`: a “kitchen sink” file with all components/styles in one place for design and QA.
 - `COMPONENT_ANALYSIS.md`: notes on what we extracted from the existing WordPress templates and why we structured components the way we did.
- Purpose:
 - This is the **layout and design blueprint** for generic/non-legal landing pages.
 - IRIS gets these as examples when we ask for new marketing/service pages (e.g., plumbing, SaaS services).

4) Existing WordPress city templates (for reference)

- Directory: `calgary/`
- We used these only as **source material** to derive patterns.
- They aren't rewritten yet, but the component system was designed to mirror their structure.

3. Standard Workflow for Using IRIS

1. Pick domain + context

- Legal role page:
 - Use `instructions legal.txt` + `paralegal01-sonnet.php` as context/examples.
- Generic landing page:
 - Use `components/README.md` + `sample-page.html` + `all-components-sample.html`.

2. Provide source content

- Role / service description document.
 - Any existing copy that must be preserved.
 - Constraints (jurisdiction, compliance, “must not say” items).
3. **Ask IRIS to generate a page**
- Feed IRIS:
 - The relevant **instructions file** (context).
 - At least one **example file** (legal page or component sample).
 - Prompt along the lines of:
 - “Generate a WordPress PHP template for [ROLE/SERVICE] following the IRIS structure and placeholders, do not invent facts, enforce hero/CTA and above-the-fold rules.”
4. **Review and correct**
- Check for:
 - All IRIS blocks present.
 - Hero + CTAs correct.
 - Placeholders wherever real data is missing (no invented testimonials/awards/prices).
 - For legal: boundaries/scope/security are accurate and non-overreaching.
 - If you see recurring issues, **change the context file**, not just the one-off prompt, so the fix generalizes.
5. **Integrate**
- For WP: drop the template into the theme and wire it as a page template.
 - For static / other stacks: map the HTML into the component system while keeping the same classes and block order.
-

4. Design / Model Strategy

- We **fix design and layout** in components and examples, and let IRIS focus on:
 - IRIS block coverage.
 - Copy, structure, and placeholders.
 - This makes the system:
 - **Flexible across models**: one LLM for structured content, another (or a human) for advanced CSS/UX.
 - **Easy to evolve**: tuning = refine instructions and examples, not retrain.
-

5. Status After One Week

Completed

- IRIS **legal instructions** (instructions legal.txt) with full block definitions and rules.
- A complete **legal example page** (paralegal01-sonnet.php) implementing IRIS end-to-end.
- A generic **component system** for landing pages:

- `components/css` for global + component styling.
- `components/partials` for modular sections.
- `components/sample-page.html` and `all-components-sample.html` as working examples.
- Documentation:
 - Component usage and placeholders (`components/README.md`).
 - Extraction rationale (`components/COMPONENT_ANALYSIS.md`).
 - **Handover doc:** `IRIS-handoff-2025-11-28.md` in the repo root.

Next recommended steps for Rahul

- Extend this beyond legal:
 - Clone `instructions legal.txt` into other domains (RevOps, marketing ops, finance) and adapt the pains/solutions/boundaries.
 - Add more role examples:
 - E.g., BDR pod page, RevOps analyst page, Marketing Ops manager, etc.
 - Start replacing legacy WordPress templates with IRIS-generated ones wired to `components/` where possible.
 - Decide how we want to centralize JS behaviours (mobile nav, FAQ, scroll animations) across pages.
-

6. Where This Lives in the Repo

- **Handover document:** `IRIS-handoff-2025-11-28.md`
 - **Legal instructions:** `instructions legal.txt`
 - **Legal example page:** `fvg-from-gpt/paralegal01-sonnet.php`
 - **Component system & docs:** `components/` (including `README.md`, `sample-page.html`, `all-components-sample.html`)
-

Revision #3

Created 28 November 2025 21:35:51 by Paddy Yoosin

Updated 28 November 2025 21:49:31 by Paddy Yoosin