

BYO-UI

Message House

A Thesis on the Future of Enterprise SaaS

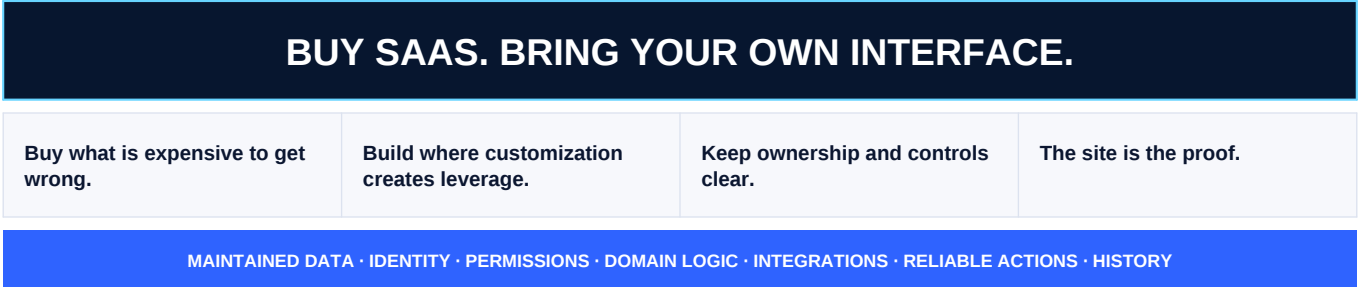
Buy SaaS. Bring your own interface.

Version 1.1 | July 12, 2026

Thesis and editorial direction: Harry Hawk

Site research, design, build, and optimization: Codex Sol 5.6 Extra High

Executive message house



Roof: Buy SaaS. Bring your own interface.

Pillar 1	Pillar 2	Pillar 3	Pillar 4
Buy what is expensive to get wrong.	Build where customization creates leverage.	Keep ownership and controls clear.	The site is the proof.
Mature SaaS can lower TCO and risk by maintaining the difficult foundation.	AI makes focused exports, artifacts, interfaces, agents, workflows, and feeds practical.	Design from the job backward and preserve ownership, approvals, provenance, and receipts.	Harry originated and directed the thesis. Codex researched, designed, built, and optimized the site.

Foundation: Maintained data, identity, permissions, domain logic, integrations, reliable actions, operational history, security, and evidence.

The thesis in one paragraph

AI has changed where software should be built. It has not erased the value of mature enterprise SaaS. Organizations still need secure data, identity, permissions, domain logic, integrations, reliable actions, operational history, and ongoing maintenance. Those responsibilities are expensive to recreate and risky to get wrong. BYO-UI separates that maintained foundation from the experience. Buy or retain SaaS for the responsibilities that are expensive to get wrong. Then use AI-assisted building to shape an export, report, artifact, application, agent, workflow, or feed around a specific person, team, or department. Keep ownership and consequential actions in the systems responsible for them.

What the site must leave people believing

1. Enterprise SaaS is not disappearing. Its durable value is becoming easier to see.
2. The native UI is important, but it does not have to be the only interface.
3. AI changes the economics of the experience layer more than the system layer.
4. BYO-UI is useful now, from a custom export to a controlled multi-system application.
5. A good implementation begins with a person, team, or department, the work to be done, and the desired outcome - not a protocol.
6. Ownership, permissions, approval, provenance, and receipts belong in the design.
7. Harry Hawk originated the thesis and can direct Codex to turn an idea into researched, working software.

Strategic foundation

Category

BYO-UI is a thesis and operating model for the future of enterprise SaaS.

It is not:

- A software product or service.
- A marketplace or vendor ranking.

- An MCP directory.
- A claim that every job belongs in chat.
- An invitation to vibe-code a replacement for Salesforce, HubSpot, or another mature system.
- An excuse to bypass permissions, approval, policy, or source ownership.

Positioning statement

For enterprise SaaS leaders, buyers, operators, and builders who need software to fit real work without recreating the systems underneath, BYO-UI separates the maintained SaaS foundation from the customer-shaped experience. Unlike full-stack vibe coding, vendor-locked interfaces, or warehouse-only customization, BYO-UI keeps durable data, permissions, logic, integrations, actions, and history in mature systems while using AI to create focused experiences around a person, team, department, job, and outcome.

Differentiated promise

Keep the enterprise SaaS foundation. Shape the interface around the work.

The shift

Old assumption: Buying SaaS means accepting its interface as the permanent boundary of what users can do.

New reality: A customer can retain the maintained SaaS foundation - its data, permissions, integrations, reliable actions, and history - while creating a better-fit experience through APIs, SQL or query access, events, SDKs, extensions, Reverse ETL, MCP, or other supported access surfaces.

BYO-UI point of view: The build-versus-buy decision has split into two decisions. Buy the system foundation when it lowers TCO and risk. Build or choose the experience that fits the job.

Alternatives and why they fall short

Alternative	What it gets right	Where it falls short
Accept the native interface as the only interface	Keeps vendor support and governance intact	Forces every person and job through the same product roadmap and interaction model
Request another SaaS enhancement	Preserves the core product	Can take quarters, compete with many customers, and still produce a generic compromise
Rebuild the entire application with AI	Maximizes freedom	Inherits security, data quality, permissions, integrations, maintenance, reliability, and compliance responsibilities
Push everything into a warehouse and BI	Enables custom analysis	Often stops at reporting and central data teams instead of serving individual jobs, actions, and workflows
Put a generic chat layer over every system	Makes questions easy to ask	Can obscure data lineage, action boundaries, interface fit, and the value of purpose-built interaction
Use BYO-UI	Keeps the trusted foundation and opens the experience layer	Requires disciplined ownership, access, approval, and provenance design

Audience hierarchy

Primary audience

Enterprise SaaS product, GTM, operations, and technology leaders.

They should see BYO-UI as a way to expand the value of existing systems without forcing every workflow into a core product roadmap or rebuilding the risk-bearing foundation.

Secondary audience

Employers, collaborators, professional peers, researchers, educators, and builders evaluating Harry Hawk's thinking and ability to direct advanced Codex work.

They should see the site as proof that Harry can originate a strategic idea, define the right constraints, direct an agent through research and implementation, challenge weak output, and turn the result into a coherent public system.

Audience translations

Audience	What they care about	Message to lead with	Proof to show
SaaS vendors	Expansion, adoption, roadmap pressure, defensibility	Let customers shape more experiences without bloating the core product	Multiple access surfaces, role-specific patterns, controlled actions
Buyers and technology leaders	TCO, risk, flexibility, lock-in	Buy the maintained foundation without making the vendor UI your permanent limit	Ownership boundaries, access evidence, availability, approval controls
Product leaders	User needs, roadmap choices, platform strategy	Treat the native UI as a strong default and the product as a foundation others can extend	APIs, events, SDKs, extensions, MCP, documented action depth
GTM and operations leaders	Answers, speed, adoption, trusted data	Unlock the technology you are already paying for around the job your team has now	Custom briefs, queues, reports, account rooms, operational workflows
Builders and architects	Access, permissions, data movement, writes	Start with the job and system ownership, then choose the narrowest sufficient access surface	Reads, writes, approvals, provenance, receipts, validation steps
Employers and collaborators	Strategic clarity, AI fluency, execution	Harry can direct Codex from thesis to research, product definition, build, QA, and communication	The working site, BYO-UI Atlas, Briefs tool, message system, and documented constraints

Core message library

Tagline

Buy SaaS. Bring your own interface.

Display rule: On constrained headline layouts, use **"Buy SaaS. Bring your own user interface."** when the added word creates a more balanced line break. This is a display-only variant; the canonical roof message remains unchanged everywhere else.

Use **user** selectively when it improves plain meaning or creates a more deliberate headline line break. Do not force it into every use of interface.

Concise economic claim

AI changes TCO. It doesn't kill SaaS.

Category descriptor

A Thesis on the Future of Enterprise SaaS

Core value proposition - 11 words

Keep the enterprise SaaS foundation. Shape the interface around the work.

Harry's first-person formulation

AI is not going to kill enterprise SaaS. It changes where customers should build. Mature SaaS gives you the secure, maintained data foundation. AI makes it practical to build the export, artifact, app, agent, or workflow your team actually needs. I call it BYO-UI.

Research-site formulation

BYO-UI separates the maintained enterprise SaaS foundation from the experience built around a person, team, or department and the work they need to do. The site provides the evidence, patterns, compositions, and planning tools needed to test that separation responsibly.

Elevator pitch

AI has changed the TCO calculus for software. It is now much easier to build a focused interface, but secure data, permissions, integrations, domain logic, reliable actions, and maintenance are still expensive to recreate. BYO-UI says: buy or retain mature SaaS for that maintained foundation, then shape the experience around the work. The result might be a CSV, an executive brief, an interactive artifact, a role-specific app, an agent, a workflow, or a controlled feed into another system.

25-word boilerplate

BYO-UI is Harry Hawk's thesis: buy maintained enterprise SaaS, then use AI to shape task-specific exports, artifacts, interfaces, agents, and workflows around the work itself.

50-word boilerplate

BYO-UI is Harry Hawk's thesis about the future of enterprise SaaS. Buy or retain mature SaaS for maintained data, permissions, integrations, domain logic, reliable actions, and history. Then use AI-assisted building to create focused exports, artifacts, applications, agents, workflows, and feeds for people, teams, departments, and real work.

100-word boilerplate

BYO-UI is Harry Hawk's thesis about the future of enterprise SaaS. AI has made focused software experiences dramatically easier to build, but it has not eliminated the value of secure data, identity, permissions, domain logic, integrations, reliable actions, operational history, and maintenance. Organizations can buy or retain mature SaaS for that foundation, then create the export, report, artifact, application, agent, workflow, or downstream feed a person, team, or department requires. The site turns the idea into practical research through the BYO-UI Atlas, interface patterns, multi-system compositions, Briefs, a glossary, Posts, and documented implementation boundaries.

Pillar 1 - Buy what is expensive to get wrong

Tension

AI makes an interface look cheap. That can make the system underneath look cheap too. It is not.

Claim

Mature enterprise SaaS can lower total cost of ownership and operational risk by maintaining responsibilities that are expensive to recreate and easy to underestimate.

Audience benefit

Organizations can move faster at the experience layer without quietly accepting responsibility for security, data quality, permissions, integration upkeep, domain edge cases, audit history, reliability, and upgrades.

Reasons to believe

- Mature SaaS maintains structured data models and operating history.
- Identity, authorization, consent, and permissions are already part of the product boundary.
- Integrations and synchronization are maintained across vendor changes.
- Domain logic and edge cases reflect years of production use.
- Reliable actions, audit trails, and operational support reduce hidden ownership.
- Buying the foundation concentrates scarce engineering effort where custom work creates more value.

Examples

- Keep customer identity, lifecycle stages, permissions, and activity history in the CRM.
- Keep attribution models, scoring, and connector health in the GTM intelligence platform.
- Keep payment state, refunds, and financial controls in the payment system.
- Keep campaign delivery and consent enforcement in the marketing platform.

Objection

"Vibe coding makes building the whole product cheap."

It makes the first visible version cheaper. A fair TCO comparison also includes authentication, permissions, data quality, migrations, monitoring, incident response, integration breakage, compliance, upgrades, and the people who own those responsibilities after launch.

Boundary

Do not claim that buying SaaS is always cheaper. Say that a responsible comparison must include the full cost and risk of operating the foundation being rebuilt.

Recommended CTA

Identify the foundation you do not want to rebuild.

Pillar 2 - Build where customization creates leverage

Tension

No SaaS roadmap can anticipate every person, question, decision, exception, or moment of work.

Claim

AI-assisted building creates the most leverage at the interface and experience layer, where a focused output can be shaped around a real job without recreating the system underneath.

Audience benefit

Individuals and teams can unlock more value from software they already pay for. Vendors can serve more use cases without turning the native product into an endless collection of exceptions.

Reasons to believe

- A useful BYO-UI can be smaller than an application.
- Modern SaaS exposes data and capabilities through several access surfaces.
- AI reduces the time needed to assemble task-specific outputs and interaction patterns.
- Narrow experiences can match the vocabulary, sequence, and decision boundary of one role.
- One maintained foundation can support many different experiences.

Examples

- A purpose-built CSV or XLSX export.
- An inline graph that answers one question.
- An executive brief assembled from current source data.
- An interactive account or campaign artifact.
- A role-specific application or operational queue.
- An agent that prepares work for approval.
- A live workflow that stays current as the SaaS data changes.
- A controlled SQL, query, or Reverse ETL feed into BI or another operational system.

Objection

"Is this just another dashboard?"

No. Reporting is one form. BYO-UI also covers artifacts, applications, agents, approvals, orchestration, and controlled action. The experience should be only as large as the work requires.

Boundary

Do not glorify custom code for its own sake. Use the native SaaS interface when it already serves the job well. Build only when a clearer, faster, or more relevant experience creates meaningful value.

Recommended CTA

Name the person, team, or department and the work. Build the smallest useful experience.

Pillar 3 - Keep ownership and controls clear

Tension

A custom interface can improve the work and still create a shadow system, unsafe write path, or second source of truth.

Claim

BYO-UI works when controls travel with capability. Ownership, permissions, approvals, provenance, and receipts are part of the interface design. In technical sections, **governance** is the combined practice of keeping those responsibilities explicit and enforceable.

Audience benefit

Teams can move beyond static reporting without losing accountability or confusing where data and decisions belong.

Reasons to believe

- The implementation sequence begins with the person, team, or department, the work, and the outcome.
- Each required read is tied to the source system that owns it.
- The experience form is selected after the job and data are understood.
- Actions are classified as read-only, draft, reversible, approval-gated, or consequential.
- Human approval remains visible where judgment or risk requires it.
- Source links, timestamps, provenance, and action receipts make the result inspectable.
- The BYO-UI Atlas, Patterns, Compositions, and Briefs make these boundaries explicit.

The implementation sequence

1. Name the person, team, or department and the work.
2. Define the outcome.
3. Identify the systems that own the required data and actions.
4. List the required reads.
5. Choose the smallest useful experience form.
6. Set the approval and action boundary.
7. Preserve provenance and receipts.
8. Validate exact access, scopes, limits, availability, and write behavior before production.

Examples

- Read CRM and product signals into an account brief; finalize outreach in the CRM.
- Draft a campaign change from current analytics; require approval before activation.
- Coordinate CRM, intelligence, and content systems in one workspace while returning each action to the responsible platform.
- Maintain a local read model for speed while showing freshness, source identity, sync status, and receipts.

Objection

"Does BYO-UI create another source of truth?"

It should not. A BYO-UI may keep a cache, read model, local preference, or draft, but the ownership contract must stay explicit. Consequential state belongs in the system responsible for it unless a deliberate migration changes that responsibility.

Boundary

Do not infer a safe write path from read access. Do not infer MCP writes from broader API writes. Revalidate plan, region, authentication, permissions, quotas, availability, and exact action behavior before implementation.

Recommended CTA

Build the brief before building the interface.

Pillar 4 - The site is the proof

Tension

AI fluency is easy to claim. It is harder to show a coherent, researched, working system produced through sustained agent direction.

Claim

BYO-UI is both Harry Hawk's thesis and a working demonstration of the direction of Codex through research, product definition, information architecture, design, implementation, testing, optimization, and public communication.

Audience benefit

Readers get a useful thesis and practical research. Professional evaluators get observable proof of Harry's strategic judgment and ability to produce with advanced AI systems.

Reasons to believe

- Harry originated the thesis and continued developing it through public writing.
- The thesis took shape while Harry supported enterprise RevSure customers balancing requests for custom enhancements against a shared product roadmap.
- Harry set the goals, constraints, corrections, examples, and editorial judgment.
- Codex Sol 5.6 Extra High performed the site's research, information design, software build, and optimization.
- The site is more than a landing page: it includes the searchable BYO-UI Atlas, full evidence profiles, filters, comparison, Patterns, Compositions, Briefs, a Glossary, How This Site Was Made, Posts, and contextual films.
- The current research release contains 158 reviewed profiles, 697 first-party source records, 37 interface patterns, and 12 multi-system compositions.
- The build and validation process checks static routes, links, evidence boundaries, interaction behavior, metadata, responsive layout, and public-package safety.

Canonical authorship lines

Short: Thesis by Harry Hawk. Site researched, designed, built, and optimized by Codex Sol 5.6 Extra High.

Extended: Harry Hawk originated the BYO-UI thesis and directed the work. Codex Sol 5.6 Extra High - not Harry - performed the site's research, information design, software build, and optimization.

Professional proof: The site demonstrates the direction of Codex from an original strategic thesis through research, product definition, design, implementation, testing, and public communication.

Objection

"If Codex built it, what did Harry do?"

Harry supplied the original thesis, strategic direction, constraints, source examples, corrections, quality bar, and final judgment. Directing the work is the capability being demonstrated. The point is not that Harry manually wrote every line. The point is that he can use Codex to turn a point of view into a rigorous public artifact.

Boundary

Never say Codex originated the thesis. Never say Harry personally performed the site's research, design, or implementation. Do not describe the production as independent of Harry's direction. Separate intellectual authorship, direction, execution, evidence, and approval clearly.

Recommended CTA

Explore the work. Judge the thesis and the execution.

Proof hierarchy

Use proof in this order. The strongest available proof should sit closest to the claim.

1. **Working behavior:** searchable BYO-UI Atlas, combined filters, comparisons, Briefs, responsive interface, captions, and accessible transcripts.
2. **Dated evidence:** first-party source records, verification dates, confidence levels, availability status, and explicit unresolved questions.
3. **Concrete examples:** exports, reports, artifacts, applications, agents, workflows, approval paths, and cross-system compositions.
4. **Visible methodology:** separation of verified facts, editorial assessments, proposed designs, and implementation caveats.
5. **Authorship transparency:** Harry owns the thesis and direction; Codex owns research, design, build, and optimization.

6. **Counts with dates:** use inventory numbers only with the current research snapshot and never imply permanent completeness.

Voice and tone - Harry Hawk, without AI ticks

The voice

Harry's writing is direct, energetic, practical, and willing to make a sharp claim. It sounds like a person who has seen a change in the market and wants other people to see it too.

Core traits:

- Lead with the tension.
- Say what is changing and what is not.
- Use plain words before architectural terms.
- Name real systems, jobs, outputs, and decisions.
- Prefer a strong point of view to a neutral summary.
- Move quickly from idea to implication.
- Use first person when Harry is speaking: "I call it BYO-UI."
- Keep the reader close to action: buy, build, test, explore, compare, decide.

Harry's natural patterns

- "AI isn't going to kill enterprise SaaS. It changes where customers should build."
- "Stop waiting for every SaaS enhancement. Build the experience your team needs."
- "Unlock the technology you are already paying for."
- "The secure, well-structured data is the base. The interface is where you can move."
- "Don't rebuild the foundation. Build on top of it."
- "I call it BYO-UI."

AI ticks to remove

Do not let the copy sound like a generic AI answer.

- No "In today's rapidly evolving landscape."
- No "It's not just X; it's Y" template repeated across sections.
- No canned transitions such as "Here's the thing," "Let's dive in," or "The bottom line is."
- No automatic three-item lists when two or four items are more honest.
- No repeated contrast formula in every paragraph.
- No buzzword piles: seamless, transformative, robust, revolutionary, game-changing, holistic, cutting-edge.
- No vague claims about "unlocking potential" without naming what is unlocked.
- No fake intimacy, invented quotations, or rhetorical questions the reader would not ask.
- No excessive em dashes, semicolons, bold fragments, or sentence fragments used only to manufacture punch.
- No pristine corporate symmetry that removes Harry's natural urgency.
- No restating the same conclusion three times with different nouns.
- No unnecessary "can help," "may enable," or "serves as" when a direct verb is accurate.
- No protocol jargon before the person, job, and outcome are clear.

Sounds like Harry / does not sound like Harry

Sounds like Harry	Does not sound like Harry
AI is not going to kill SaaS. It changes where you should build.	In today's rapidly evolving AI landscape, organizations must reimagine their software ecosystems.
Unlock the technology you are already paying for.	Unlock transformative value through a seamless, holistic experience layer.
Buy the secure data foundation. Build the experience your team needs.	Leverage robust enterprise capabilities to enable bespoke user-centric journeys.
No SaaS roadmap can build every interface for every customer.	Modern businesses increasingly require enhanced flexibility and personalization.
Start with the person, team, or department, the work, and one useful outcome.	Organizations should embark on a comprehensive stakeholder-centric transformation initiative.
I call it BYO-UI.	This innovative paradigm may be referred to as bring-your-own-user-interface architecture.

Channel adjustments

Channel	Voice treatment
Home and Thesis	Strong point of view, short contrasts, concrete explanation
BYO-UI Atlas and How This Site Was Made	Precise, neutral, dated, evidence-first
Patterns, Compositions, and Briefs	Practical, sequential, action-oriented
Posts and LinkedIn	First person, more provocative, recognizably Harry
About	Transparent, confident, specific about authorship and Codex direction
Metadata and accessibility copy	Clear and literal; personality never reduces comprehension

Posts route

Headline: The Pundits claiming SaaS Death are wrong.

Supporting copy: This is where the thesis started and grew: SaaS is not going away, but UI lock-in is. These are Harry Hawk's original posts, graphics, and source links.

Use **thesis**, **case**, **idea**, or **position** when naming the body of work. Avoid using **argument** as a noun for the thesis because it implies a fight. Natural constructions such as “Harry argues that SaaS gains value from AI” remain appropriate.

Terminology and claims guardrails

Preferred terms

- Buy SaaS. Bring your own interface.
- Enterprise SaaS foundation.
- Maintained SaaS foundation.
- Customer-shaped experience.
- Interface or experience layer.
- Person, job, and outcome.
- Required reads.
- Approval and action boundary.
- Provenance and receipts.
- AI-assisted building.
- Vibe coding, when speaking in Harry's public voice and when the system boundary is stated.

Use with precision

Term	Rule
BYO-UI	Covers exports, artifacts, feeds, interfaces, agents, and workflows, not only screens
SaaS	Name the durable foundation being purchased, not a generic subscription
Vibe coding	Use for the high-leverage experience layer, not as a substitute for maintained data, permissions, security, ownership, or controls
MCP	Present as one access surface alongside APIs, SQL, events, SDKs, and extensions
Read/write	Classify the exact surface; do not infer MCP writes from API writes
Ready	Tie readiness to evidence, access, action depth, availability, and validation
Complete	Avoid for changing research inventories; use reviewed or current
Best	Do not use for platform quality or procurement recommendations
Secure	Attribute to documented controls or use as a responsibility, not an unsupported certification

Claims discipline

- Say mature SaaS **can** lower TCO and risk; do not say it always does.
- Say a capability is documented, reviewed, tested, proposed, or unresolved.
- Keep platform evidence first-party wherever possible.
- Date inventory counts and verification claims.
- Do not turn Atlas order into a vendor-quality ranking.
- Do not present a proposed interface as a vendor-shipped product.
- Preserve the RevSure professional-relationship disclosure wherever RevSure is discussed as an example.
- Do not expose private research databases or source manifests through the public build.
- Use **governed** only when governance is defined or technically necessary. In general-audience copy, name the actual responsibility: maintained data, permissions, integrations, ownership, controls, reliable actions, or history.
- Preserve historical titles and direct quotations exactly, even when they use earlier language such as "Buy the system."
- The Message House governs editorial and interface copy. Platform profiles, pattern records, composition records, glossary definitions, directory rows, and A-Z entries retain their research-record language.

CTA hierarchy

The site promotes Harry's work through usefulness and proof, not a sales funnel.

Primary CTAs

- Read the thesis.
- Search the Atlas.
- Build a brief.
- Search the patterns.

Secondary CTAs

- Compare platforms.
- Inspect the evidence.
- Explore multi-system compositions.
- Read the original writing.
- Learn how the site was made.

Participation CTAs

- Test the thesis.
- Build an implementation.
- Cite the work.
- Publish what worked and what failed.
- Suggest a correction with first-party evidence.

Avoid

- Book a demo.
- Contact sales.
- Hire me.
- Request consulting.
- Download the database.
- Claim your listing as a lead-generation device.

Professional credibility should be obvious from the quality, transparency, and scope of the work. It does not need a conversion form.

Route activation map

Route	Job of the page	Message to lead with	Proof or action
Home	Make the thesis understood in seconds	Buy SaaS. Bring your own interface.	Foundation diagram, TCO film, Atlas and thesis CTAs
Thesis	Explain the economic and architectural logic	AI changes TCO. It does not kill SaaS.	Tenets, objections, practical test, Test it yourself
BYO-UI Atlas	Determine whether a platform is practical to build on	Find the SaaS platforms you can build on	Search, filters, comparison, profile sources
Patterns	Plan backward from a person, team, or department and the work	The interface should emerge from the work	Search and filters; jobs, reads, artifacts, approvals, guardrails, and MVPs
Compositions	Coordinate systems without creating shadow ownership	One experience can span systems while each retains responsibility	Cross-system reads, approval gates, action return paths, receipts
Briefs	Turn the thesis into a bounded build brief	Name the person, team, or department, outcome, systems, experience, and action boundary	Generated brief, RepSure example, and profile links
Glossary	Make the operating language usable	Shared terms reduce ambiguous architecture and unsafe assumptions	Searchable definitions and source routes
How This Site Was Made	Explain the research record and production process	Harry owns the thesis and direction. Codex performed the research, design, build, testing, and optimization.	Current counts, process, first-party sources, dates, confidence, caveats
Posts	Preserve Harry's original public voice and idea development	The pundits claiming SaaS death are wrong	Original graphics, essays, dates, summaries, source posts
About	Explain purpose, authorship, production, and professional context	Harry owns the thesis and direction. Codex built the site.	Bio, production credit, principles, citation, disclosures, Message House PDF

Site-alignment and maintenance checklist

Message House v1.1 is implemented across the site's non-listing copy. Use this checklist to keep later changes aligned.

Priority 0 - authorship accuracy

- Use the canonical short credit in persistent site production copy.
- Use the extended credit on About.
- Explain that directing Codex from thesis through public artifact is part of Harry's demonstrated work.
- Keep Harry as thesis author and editorial director; do not describe him as the site's researcher, designer, or developer.

Priority 1 - message hierarchy

- Keep "Buy SaaS. Bring your own interface." as the live roof message and "AI changes TCO. It doesn't kill SaaS." as the concise economic claim.
- Make the TCO and risk logic visible before protocol details.
- Make the person, job, and outcome visible before access surfaces.
- Treat the site itself as proof without interrupting the reader's primary task.

Priority 2 - voice

- Review all non-listing copy against the Harry voice rules.
- Remove AI ticks, generic boilerplate, buzzword stacks, and repeated rhetorical templates.
- Preserve Harry's original post titles and quotations exactly.
- Use first person only where Harry is speaking; keep research copy neutral and literal.

Priority 3 - proof and restraint

- Put dated proof close to strong claims.
- Qualify TCO claims and avoid universal economics.
- Replace permanent-sounding completeness claims with reviewed or current.
- Preserve visible evidence, availability, approval, and implementation boundaries.

Message testing questions

Use these questions when reviewing future copy or testing it with readers.

Clarity

- Can the reader explain BYO-UI after ten seconds?
- Do they understand what is being bought and what is being shaped?
- Do they know that BYO-UI can be smaller than an application?

Relevance

- Does a SaaS vendor see roadmap and expansion implications?
- Does a buyer see flexibility without casually accepting system risk?
- Does an operator see a path from current tools to a better-fit experience?
- Does a professional evaluator understand what Harry did and what Codex did?

Value

- Does the copy make the TCO and risk boundary concrete?
- Does it show useful outcomes beyond generic reporting?
- Does it make existing SaaS investment feel more valuable rather than obsolete?

Differentiation

- Is BYO-UI clearly different from full-stack vibe coding?
- Is it clearly different from a generic chat layer or MCP directory?
- Is Harry's ability to direct Codex demonstrated with working evidence rather than asserted?

Friction and trust

- Are overclaims qualified?
- Are sources, dates, and unresolved questions visible?
- Are ownership, approvals, provenance, and receipts understandable?
- Is the authorship split unambiguous?

Editorial acceptance checklist

A new copy unit is ready only when all applicable statements are true.

- It advances the thesis or helps someone plan an implementation.
- It sounds like Harry where the point of view is Harry's.
- It does not contain generic AI ticks.
- It names a person, job, output, system responsibility, or decision when specificity is needed.
- It distinguishes verified fact, editorial assessment, proposed design, and unresolved question.
- It uses the correct authorship and production attribution.
- It does not imply a product sale, paid ranking, or procurement recommendation.
- It is concise enough to scan and detailed enough to trust.
- Its CTA matches the reader's job on that page.
- It can survive being quoted without losing its intended boundary.

Source and practice appendix

Harry Hawk voice sources

- Harry Hawk, "AI has changed the TCO calculus for build/buy decision making," LinkedIn, July 2026: https://www.linkedin.com/posts/harryahawk_ai-has-changed-the-tco-calculus-for-build-share-7480386638221598720-lu3h/
- Harry Hawk, "Stop begging for new SaaS enhancements - vibe code your own," LinkedIn, July 2026: https://www.linkedin.com/posts/harryahawk_stop-begging-for-new-saas-enhancements-share-7478577692859187200-A0pK/
- Harry Hawk, "AI isn't going to kill enterprise SaaS," LinkedIn, June 2026: https://www.linkedin.com/posts/harryahawk_ai-isnt-going-to-kill-enterprise-saas-ai-share-7476394470649905153-ph0C/
- Harry Hawk, "RevSure plus MCP is the ultimate GTM unlock," LinkedIn, June 2026: https://www.linkedin.com/posts/harryahawk_revsure-plus-mcp-is-the-ultimate-gtm-unlock-ugcPost-7471271984555925504-HQqH/
- Harry Hawk, "Don't wait for companies to build an app - build it yourself," LinkedIn, June 2026: https://www.linkedin.com/posts/harryahawk_amazing-dont-wait-for-companies-to-build-ugcPost-7467048486824357889-oJCu/
- Harry Hawk, "The future of B2B SaaS is BYO-UI," LinkedIn, April 2026: https://www.linkedin.com/posts/harryahawk_saas-its-not-going-away-but-neither-is-share-7445355381020798976-O2QM/

Product marketing and copy practice

- April Dunford, positioning for B2B technology companies: <https://www.aprildunford.com/>
- Product Marketing Alliance, Messaging Framework: <https://www.productmarketingalliance.com/content/files/2025/02/Messaging-framework.pdf>
- Wynter, B2B Message Layers Framework: <https://wynter.com/post/b2b-message-layers-framework-wynter>
- Wynter, B2B Vendor Differentiation Survey: <https://wynter.com/research/b2b-vendor-differentiation-survey>
- Nielsen Norman Group, "Be Succinct! Writing for the Web": <https://www.nngroup.com/articles/be-succinct-writing-for-the-web/>

Internal research inputs

- BYO-UI site copy concordance, 917 distinct non-listing copy units, July 11, 2026.
- BYO-UI site copy review and prioritized rewrite queue, July 11, 2026.
- Current BYO-UI thesis, BYO-UI Atlas, Patterns, Compositions, Briefs, Glossary, How This Site Was Made, Posts, and About copy.
- Current reviewed research snapshot: 158 platform profiles and 697 first-party source records, verified July 10, 2026.

Canonical credit: Thesis by Harry Hawk. Site researched, designed, built, and optimized by Codex Sol 5.6 Extra High.