Lyriks.io

The coherence layer for AI-accelerated software.

Pitch Deck · 2026

CONTEXT

The economics of building software **has changed.**

BEFORE

Margin = revenue - headcount

NOW · 2026

Margin = revenue - headcount **- tokens**

Volatile · Opaque · Invisible until invoice

Worldwide GenAI costs, 2025

$644B

+76% year over year.

$13	+75%	84%
per dev / day	Comp. LLM costs YoY	devs use AI

Sources: IDC 2025, Anthropic enterprise data, Stack Overflow 2025, a16z

ADOPTION

AI Adoption is no longer optional.

It's the new baseline.

84%

of developers use AI dev tools today

Stack Overflow Survey 2025, 49 000+ respondents

+75%

growth in company LLM budgets per year

a16z survey of 100 enterprise CIOs, 2025

90%

of company engineers will use AI by 2028

Gartner, July 2025 (revised up from 75 %)

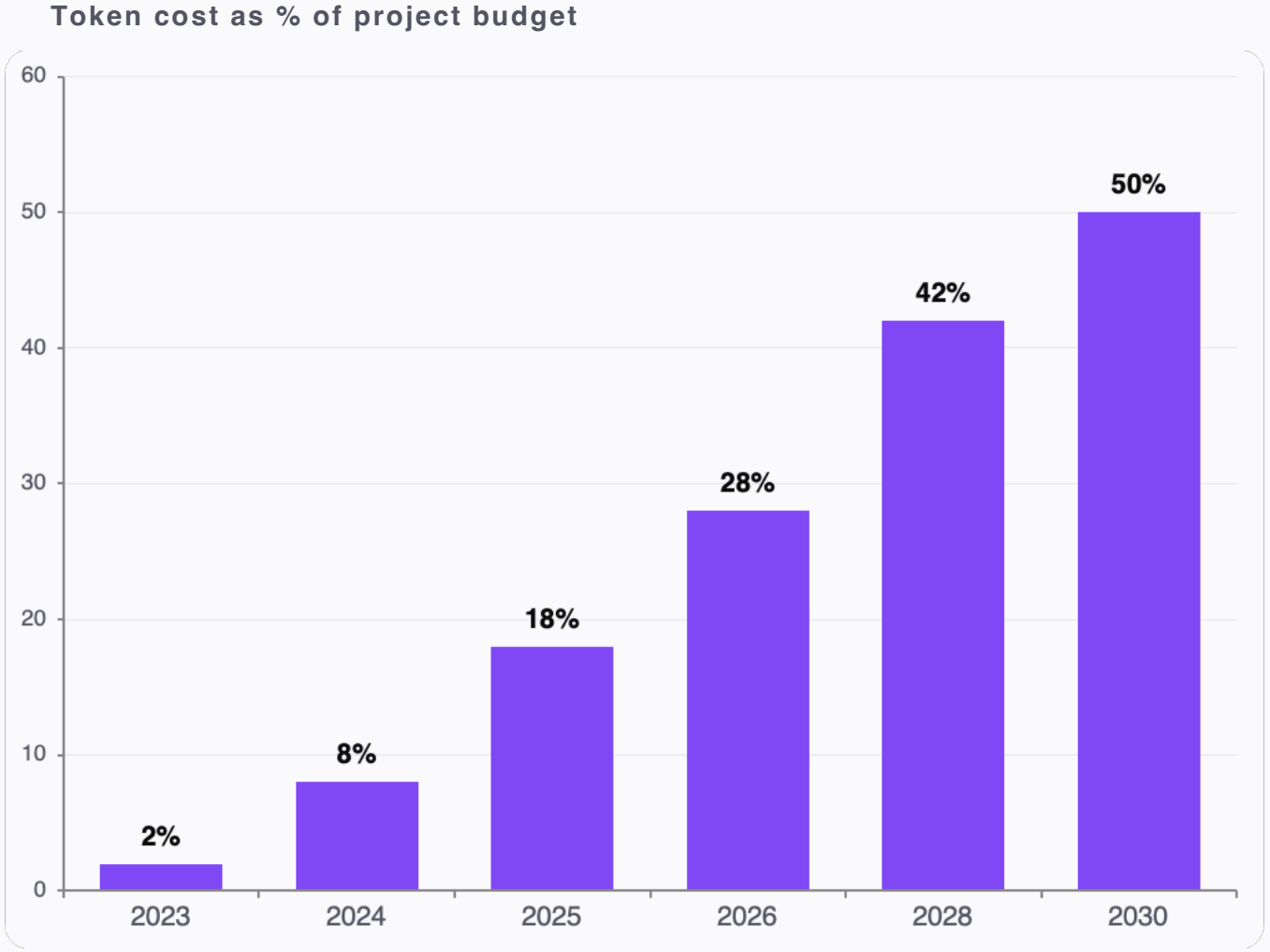
PROJECTION

Within 5 years, AI tokens will be **a major line item in project budget.**

Up to

50%

of project budget in tokens by 2030



Sources: Gartner 2025, Anthropic 2026

THE STAKES

This is no longer an IT issue. It's a **margin** problem.

Forecast

Predict the variable cost line before the period closes

Control

Set budgets and guardrails without punishing the best people

Protect

Preserve margin deterministically, not by chance

THE QUESTIONS

Three questions
every company must answer.

01

How do we reduce token cost?

Without slowing engineering down. Without sacrificing quality.

- Less context bloat per prompt
- Fewer iterations to converge
- Better specifications upstream

02

How do we get visibility on what AI spends and ships?

Per-team, per-project, per-feature.
Tied to outcomes, not to volume.

- Spend attribution by domain
- Quality measured against specs
- Coherence as a leading indicator

03

How do we scale from MVP to industrial-grade with AI?

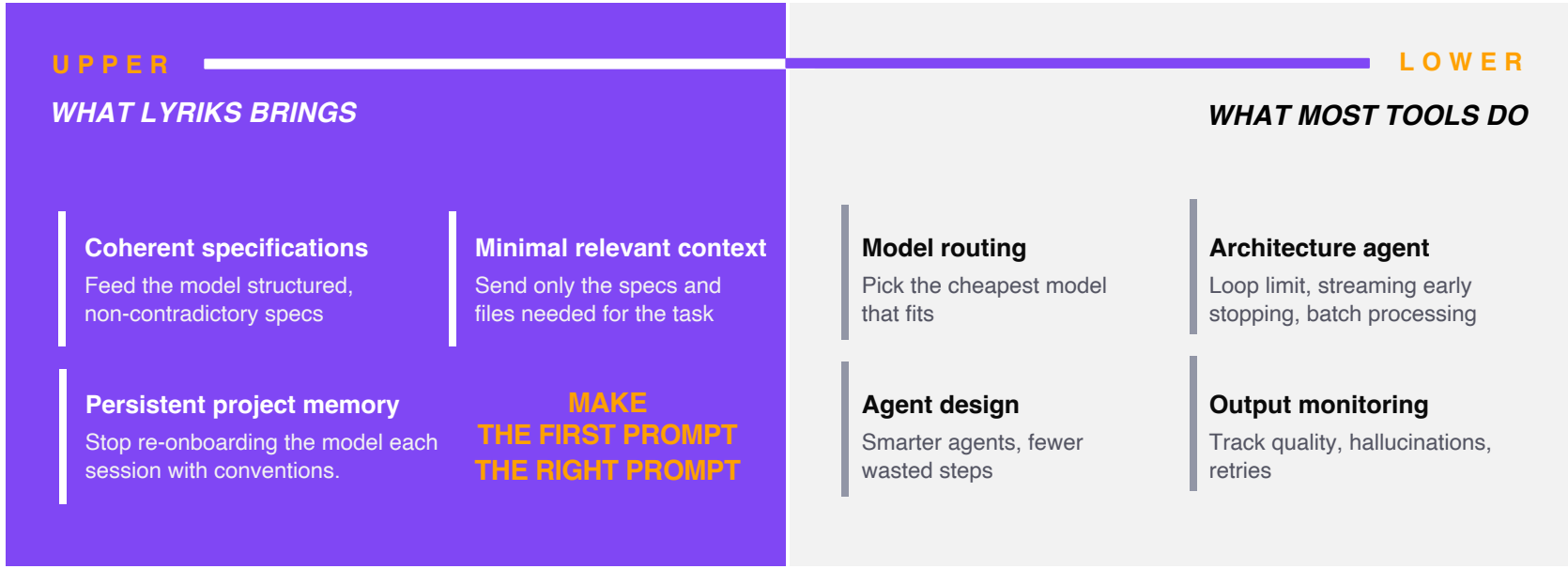
From prototype quality to production-grade software, without rewriting from scratch.

- Coherence holds as the system grows
- Real-time collaboration on shared code
- Code-to-spec keeps legacy in the loop

THE LANDSCAPE

We bring the Solution on the upper funnel.
While today's tools optimize downstream.

Lyriks ensures specifications are coherent before a single token is spent. Downstream tools fight the symptoms; the root cause is upstream: incoherent, fragmented specs fed to the model.



OUR APPROACH

Lyriks is the first software specification coherence engine.

THE OLD WAY

Fragmented knowledge .

- ✗ Knowledge in people's heads
- ✗ Scattered across documents
- ✗ Conflicts appear on merging
- ✗ Days lost writing specs
- ✗ AI hallucinates from poor input

THE LYRIKS WAY

One source of truth.

- ✓ Specifications live in Lyriks
- ✓ Generated automatically, no writing
- ✓ Feature-by-feature decomposition by design
- ✓ Misalignments caught before a single line of code is written
- ✓ MCP-connected to Claude Code, Cursor, …

HOW IT WORKS

From specifications to shipped code, with coherence at every step.

01
Ingest
Import specifications, manually or via MCP from existing tools

02
Complete
Fill in missing pieces. Lyriks outlines where coherence breaks

03
Check
Each increment is validated against the existing. No features held together with duct tape.

04
Push
Push specs to AI dev tools (Claude Code, Cursor, Copilot) via MCP

05
Analyze
Post-build analysis verifies what was specified was implemented

06
Adjust
Discrepancies pop **up** visually

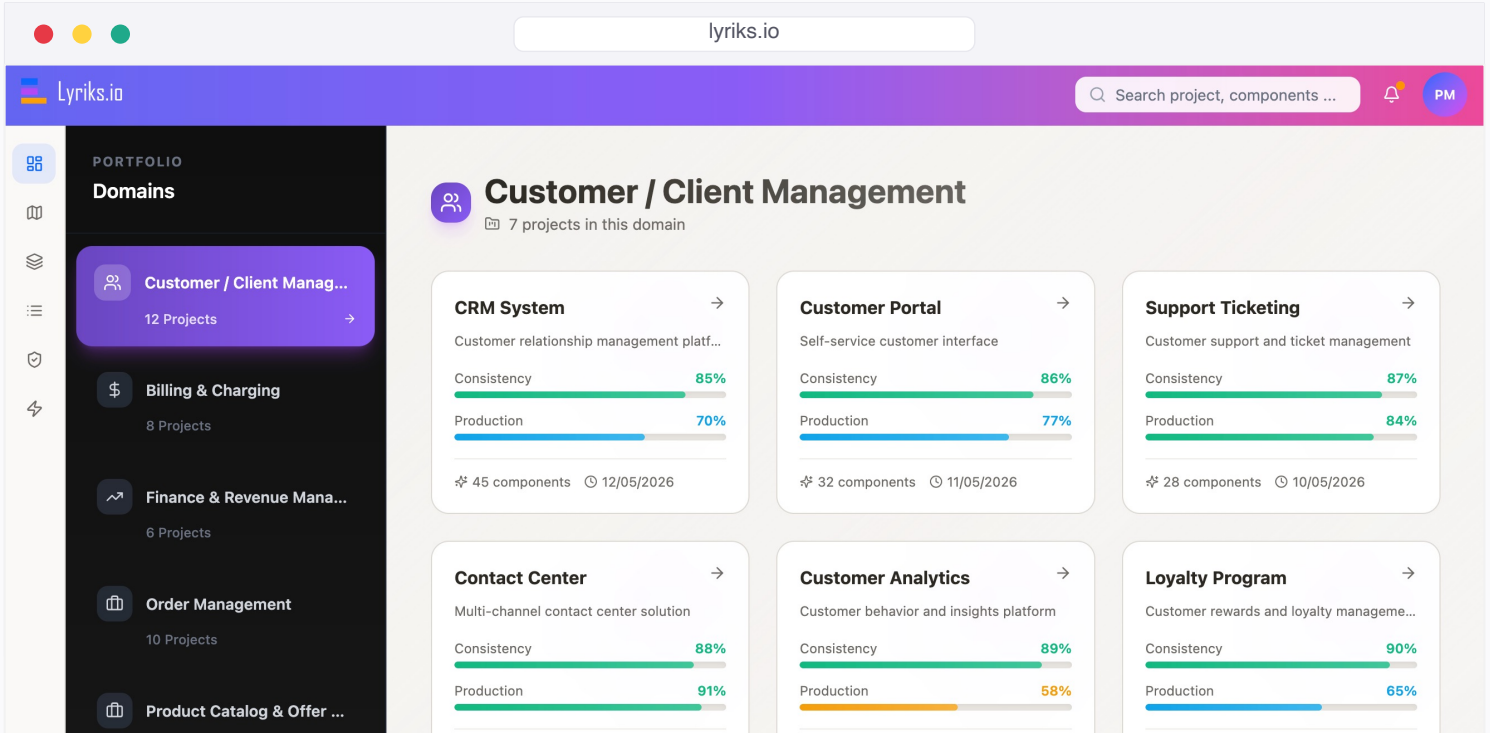
PRODUCT TOUR

A walk through Lyriks, **from design to production.**

7 screens. The full coherence loop.

PRODUCT TOUR · 01 / 7

Navigate your domains and their projects.



STEP 01

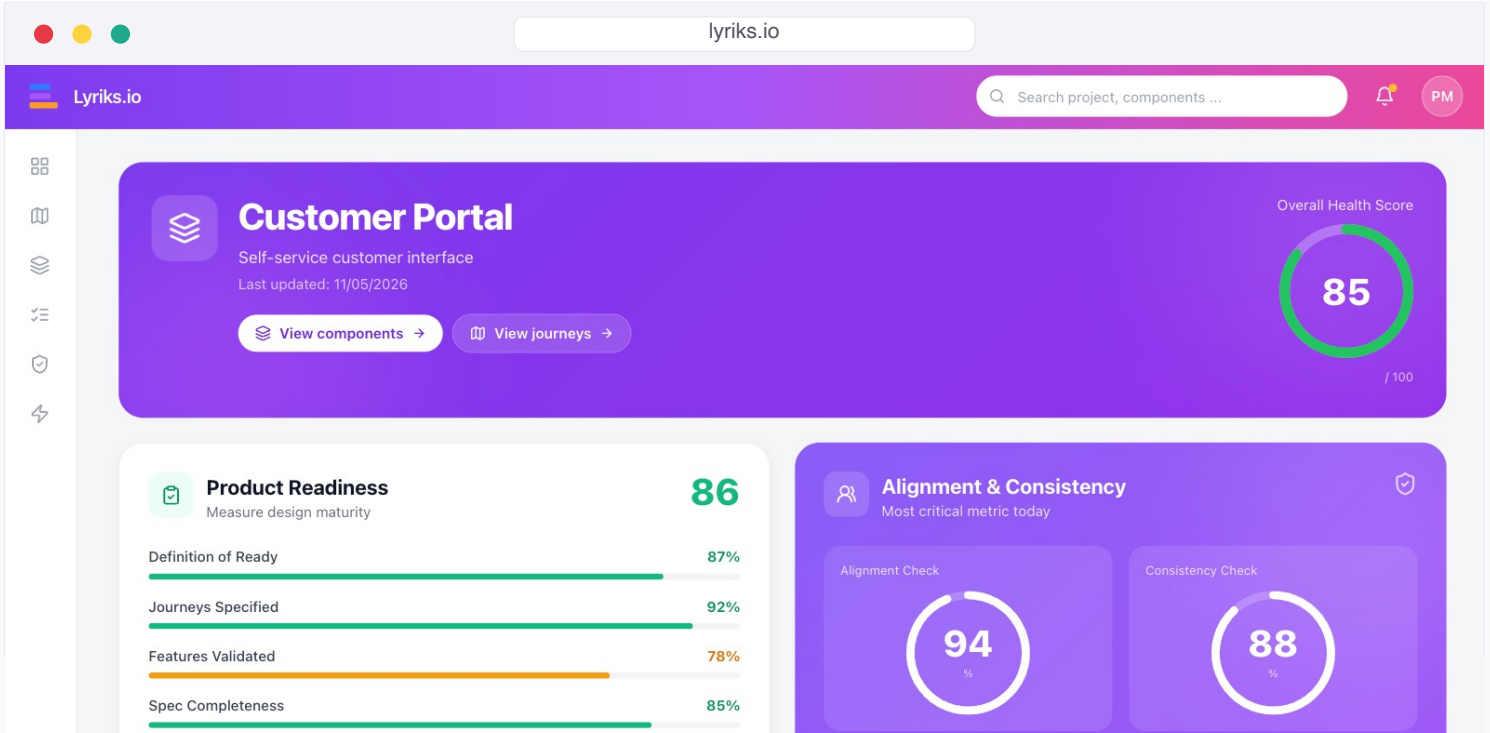
One workspace, many projects.

- Each project groups related domains
- Domain health visible at a glance
- Drill down with one click

Procurement and engineering see the same picture.

PRODUCT TOUR · 02 / 7

A full dashboard for every project.



STEP 02

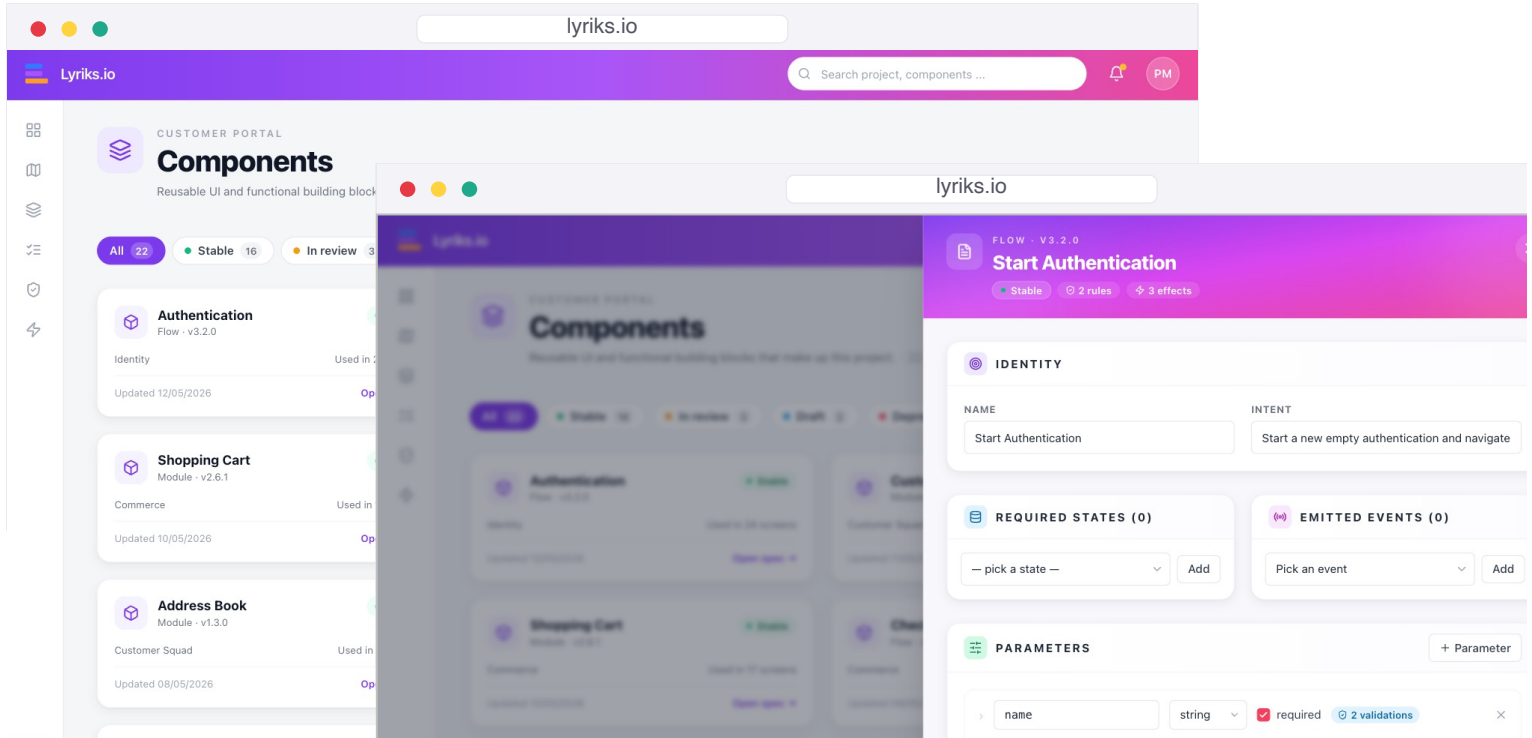
An honest picture of the current state.

- Features, journeys, data flows
- Coherence score per domain
- Live activity from the whole team

Make software health a metric, not a guess.

PRODUCT TOUR · 03 / 7

Features, rules and data…
All traced.



STEP 03

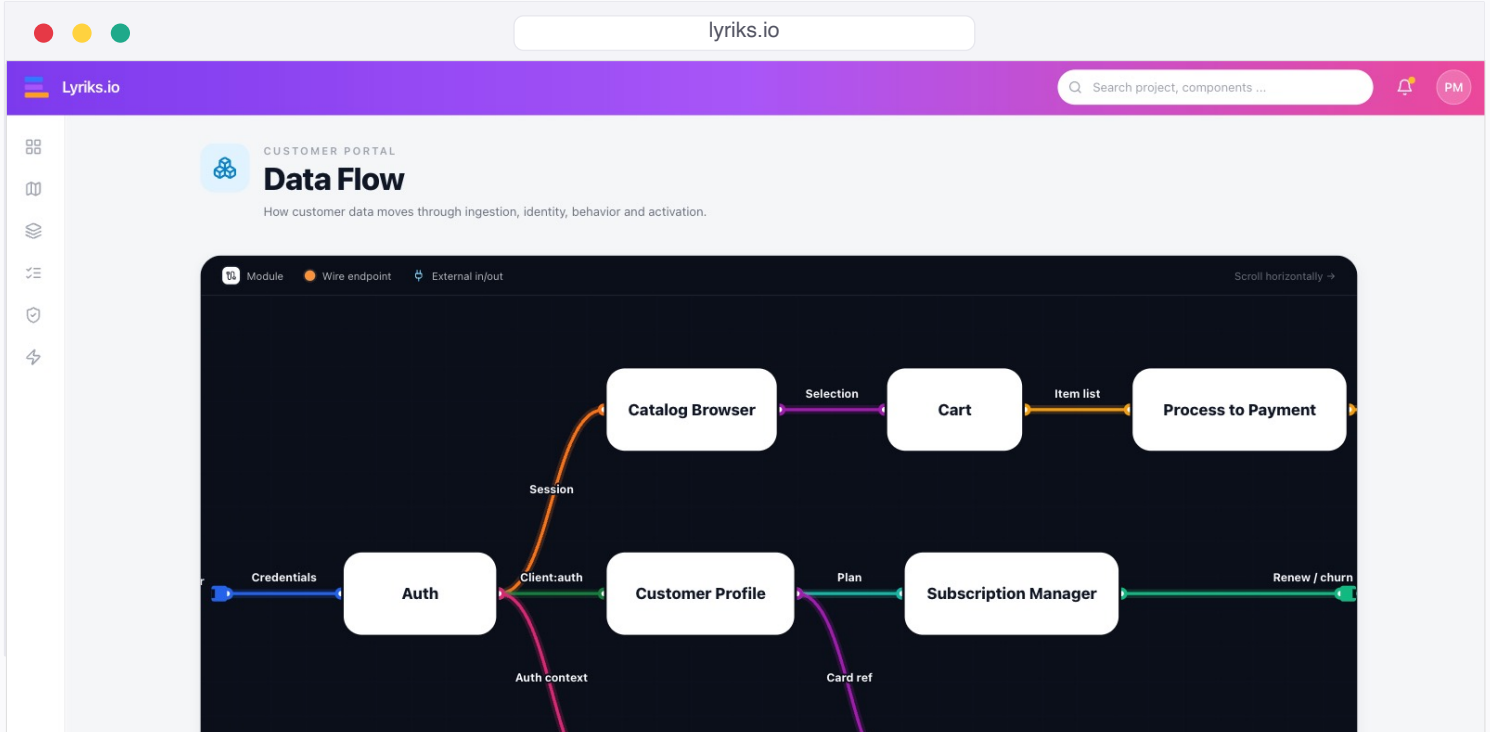
Features, rules and data logic visually accessible

- Business rules attached to each feature
- Inbound and outbound data per component
- Impact analysis in seconds, not days

Ask «what is related to ?» and get an answer.

PRODUCT TOUR · 04 / 7

Visualise how data flows **across your system.**



STEP 05

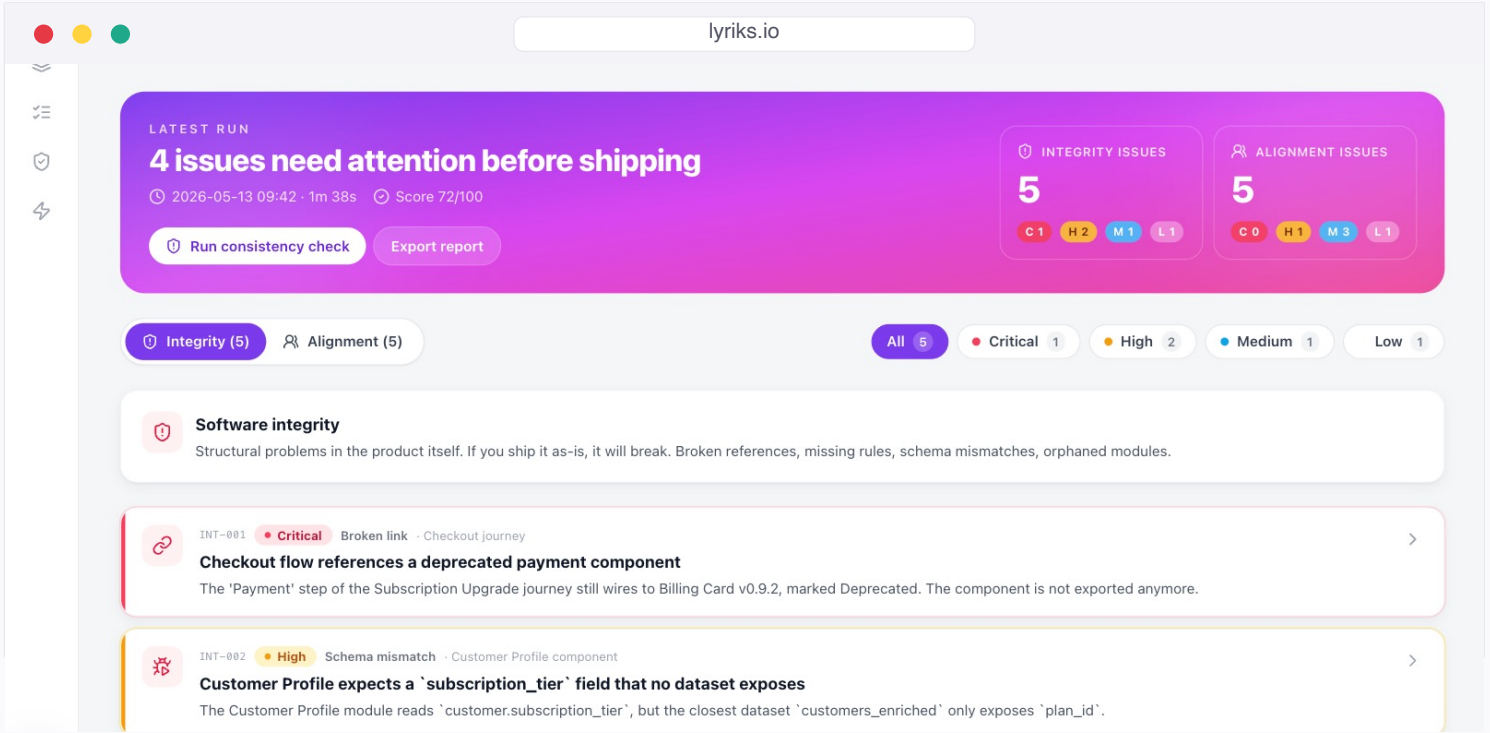
Where data comes from. Where it goes.

- Inbound and outbound dependencies
- Click any edge to see the detail
- Spot orphaned data and missing sources

Architecture decisions based on facts, not memory.

PRODUCT TOUR · 05 / 7

Two kinds of coherence issues, caught before they ship.



STEP 06

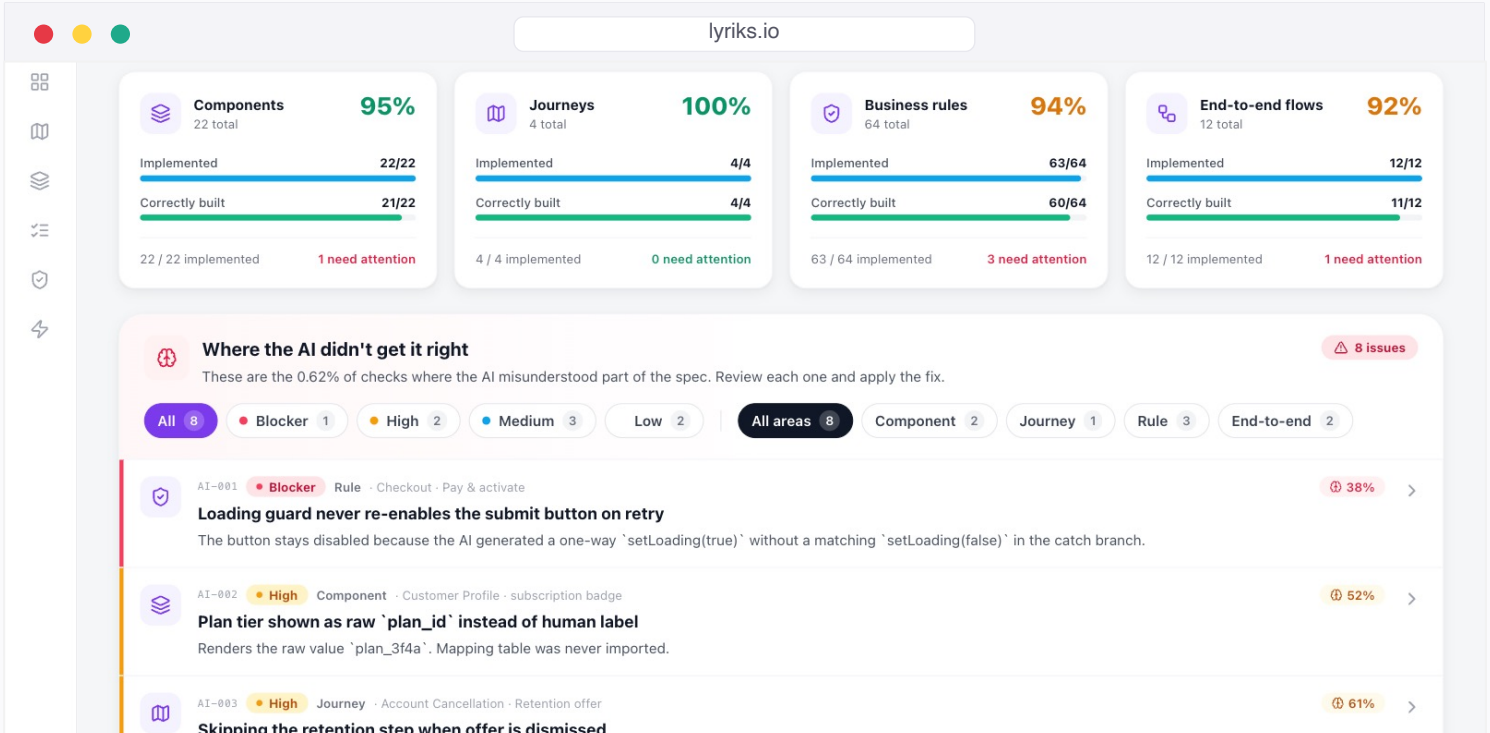
Catch what humans miss. Both detected with a one-click fix.

- Integrity: structural bugs in the spec
- Alignment: silent conflicts between contributors

Hallucinations die here. Before any AI sees the spec.

PRODUCT TOUR · 06 / 7

Compare code to spec.
Identify drift. Adjust.



STEP 08

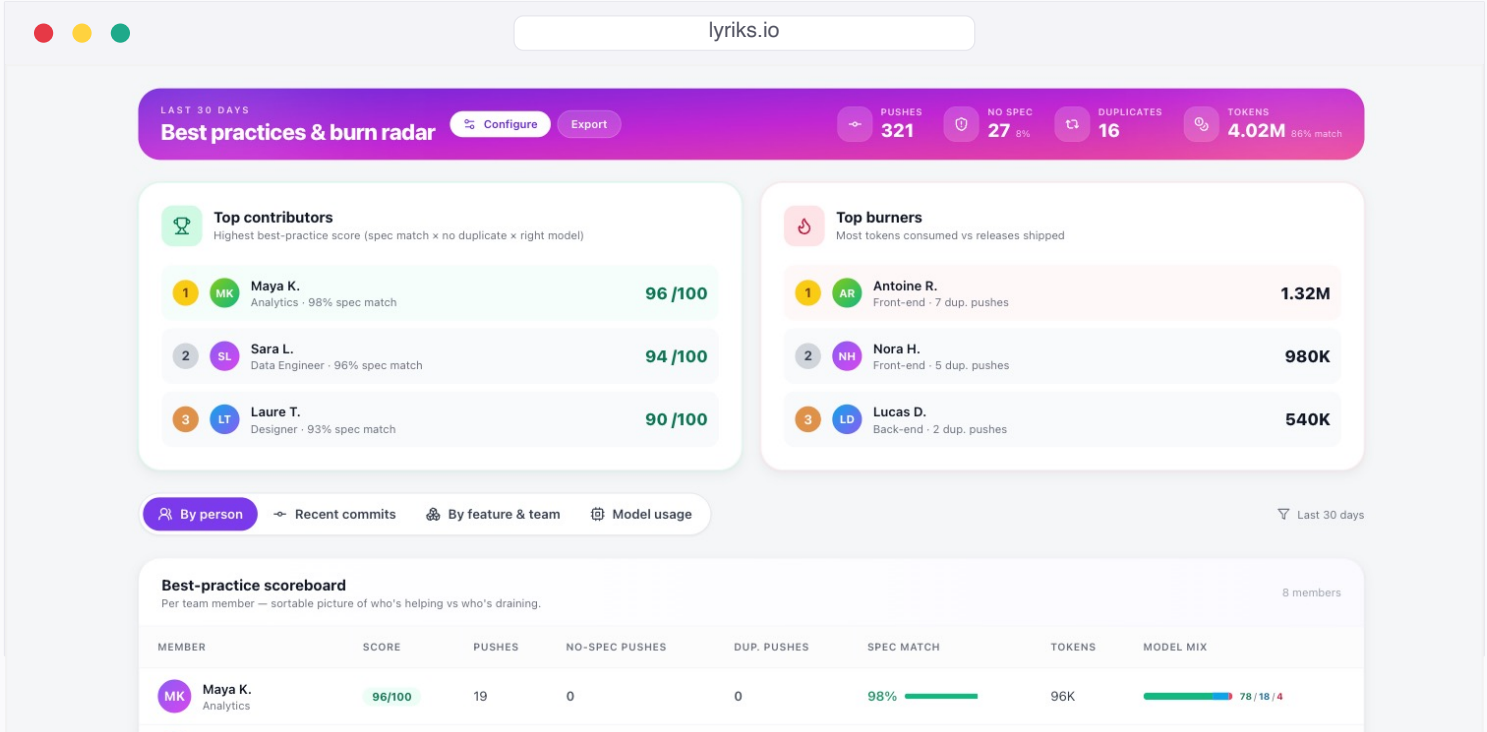
Close the loop.

- Side-by-side spec vs built code
- Drift highlighted automatically
- Regenerate with one click

Spec is the ground truth, not wishful thinking.

PRODUCT TOUR · 07 / 7

Track who follows the best practices, And who burns tokens.



STEP 09

Visibility on every contributor.

- Best-practice score per dev and team
- Cost mapped to features and outcomes
- Model choice vs task complexity, flagged

Reward good practice. Detect waste. Configure your own rules.

WHAT'S NEW

Two breakthroughs
That no other tool delivers today.

Real-time collaboration

on shared code.

Multiple developers, or multiple AI agents, work on the same codebase simultaneously, incrementally, even on legacy systems.

No more merge hell. No more rework. No more parallel branches diverging.

Code-to-Specs.

Reverse engineering, done properly.

Feed Lyriks an existing codebase, written by humans or generated by AI, and it produces a coherent specification document.

Features. Business logic. Domain rules. Journey visually, ready to extend.

THE BENEFIT

Targeted impact:
at least 30 % token reduction.

-99% hallucination

Specification coherence eliminate
the ambiguity that drives AI to make things up

-60% scoped context

Not just compressed. Lyriks gives the AI exactly
what's needed for the task at hand. Nothing else.

Single Source of Truth

One source of truth, readable and writable by every
stakeholder: PMs, devs, architects, and the AI.

Live collaboration - 0 rework

Merge conflicts and parallel rewrites disappear.
Code converges instead of diverging

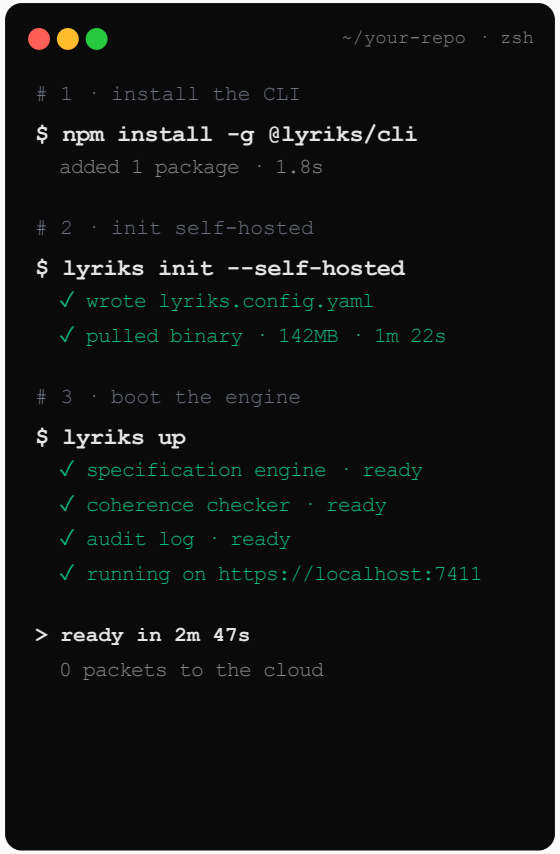
DEPLOYMENT

On-prem in 3 minutes. One command.

No cloud lock-in. No data leaving your VPC.
Run Lyriks where your code lives.

- ✓ Your repo, your tenant, your rules
- ✓ Zero data egress · prompts stay local
- ✓ Air-gapped friendly · no phone-home
- ✓ Same binary on Mac, Linux, Docker, k8s

00:03:00 *avg install time*



Lyriks.io

NATIVE MCP

One protocol.
Zero vendor lock-in.

MCP is an open standard.
Adopting Lyriks doesn't lock you into our stack.
Your spec exports to plain JSON, anytime.

MCP-COMPATIBLE · SINCE LAUNCH

```
mcp.json

// drop into any MCP-aware tool

{
  "mcpServers": {
    "lyriks": {
      "command": "npx",
      "args": ["-y",
        "@lyriks/mcp-server"],
      "env": {
        "LYRIKS_TOKEN":
        "${env:LYRIKS_TOKEN}"
      }
    }
  }
}
```

CC **CLAUDE CODE** NATIVE · 30S SETUP	>_ **CURSOR** NATIVE MCP · 1 MIN	Wf **WINDSURF** NATIVE MCP · 1 MIN
Cl **CLINE (VS CODE)** NATIVE MCP · 1 MIN	Cn **CONTINUE.DEV** NATIVE MCP · 1 MIN	Co **GITHUB COPILOT** NATIVE MCP · 2 MIN

Lyriks.io

INTEGRATION · CURSOR

Live in Cursor in 60 seconds.

Cursor speaks MCP natively. No plugin, no SDK, no rebuild.
Add one config block and your AI reads the spec.

1 **Open the MCP config**
Cursor → Settings → MCP, or edit ~/.cursor/mcp.json directly.

2 **Paste the Lyriks block**
One server entry. Your token as an env variable. Nothing else.

3 **Reload Cursor**
The Lyriks tools appear in the agent. The AI now reads your spec.

Works the same on Cursor SaaS and Cursor self-hosted. **Your code and prompts never leave your environment.**

FOUNDATION

10 years of research.
3 years in production.

Lyriks is the result of a decade of scientific work at the intersection of AI, mathematics and formal methods, with top-tier research teams and multiple peer-reviewed publications. For the past 3 years, applied concretely on industrial software projects of all sizes.

CNRS | DARPA | HARVARD | ENS

WHY IT MATTERS

Categorical Graph Rewriting

Every edit is a formal, typed operation. The math guarantees that parallel changes can be merged without conflict.

5+ people edit at the same time. No friction. No drift.

Built for AI-speed

Delta computation in minimal time. No global recalculation.

Scales with high-frequency AI artifact streams.

Petri-net compliant

The same model that specs your system can simulate it.

Detect deadlocks at spec level, not in production.

Lyriks.io

We build the coherence layer
for the AI-software era.

Let's talk.