

PRODUCT BROCHURE

Your Enterprise Software Factory

Describe what your business needs — in plain English. Codalio turns it into **specs, plans, and production-ready code**, through a fully automated agentic pipeline your team commands.

SECURE BY DESIGN

SCALES WITH REAL USERS

CODE YOU OWN

Idea → staging in a day

Not six months of scoping followed by a \$20M question mark.

9-station agentic flow

Plan, develop, test, debug, review, compliance-check, commit, merge — configured automatically.

100% traceable

Every line of code maps to a requirement, a test, and a compliance check.

"We are at the dawn of something real — one person shipping at a scale that used to require a team of twenty."

GARRY TAN · CEO, Y COMBINATOR — ON THE RISE OF THE SOFTWARE FACTORY

HOW IT WORKS

From business intent to production code.

The bottleneck was never typing speed — it's translating what the business wants into what engineering builds. Codalio automates the whole chain:

English prompt



PRD



Spec & scope



Code you own



Deploy & scale

01

Describe

State the objective in plain English: "Cut customer onboarding time by 25%."

02

Specify

Codalio asks a few sharp questions, then generates the PRD, technical spec, and acceptance criteria.

03

Plan

It reads your existing codebase and architecture — and plans within them. Scope and cost visible up front.

04

Build & ship

The factory runs. Your engineer approves the final PR. Humans hold the gate.

The factory floor — configured automatically, for every piece of work

No hand-built workflows — the orchestration layer generates the agentic flow itself. Agents work from shared context, not a prompt loop that forgets what it built yesterday.

1

Plan

Architecture & implementation plan vs. your existing systems

2

Develop

Proven patterns + AI for the code that differentiates you

3

Write tests

Coverage built from spec acceptance criteria

4

Run tests

Full suite executed, results captured

5

Debug

Failures diagnosed & fixed in-loop

6

Review

Quality, security & architecture checks

7

Compliance

Validated against your regulatory rules & standards

8

Commit

PR opens with full end-to-end traceability

9

Merge

Your engineer reviews, approves & ships

Even a CEO can describe what they want — and it's automatically scoped, developed, tested, debugged, and reviewed.

Your developers do one thing: review the pull request and merge. Codalio is the intent layer between business need and code — the gap between the boardroom and the codebase, closed.

Orchestration layer

What the best builders assemble by hand, Codalio generates automatically — roles, hand-offs, and review gates per work item.

75-85%

Abstraction layer

of enterprise code repeats known patterns. We instantiate them from pre-validated components — AI writes only what's new.

~98%

Living spec — your DNA

Edge cases sync back into the spec, so code and requirements never drift. Regenerate nearly your entire system from the spec alone.

The old way is broken. The factory fixes it.

95%

of enterprise GenAI pilots delivered zero measurable P&L impact — MIT Project NANDA, 2025

~80%

of technology projects are derailed by scope creep in the intent-to-spec translation layer

\$2.41T

annual cost of poor software quality in the U.S. alone — CISQ, 2022



Speed to decision

Six-month scoping becomes a same-day answer — with working software attached. Decide before you commit millions.



Speed to production

Quarters compress to days. Projects your roadmap couldn't reach become feasible.



Capacity, not headcount

Engineers stop translating and boilerplating — and start deciding, reviewing, shipping.



Governance you can defend

Every change traces to a requirement, test result, and compliance check. Drift is visible immediately.



Software that fits

Stop paying for 100% of a SaaS platform and using 5%. Build exactly what your business needs.



An asset you own

Your spec DNA — requirements, decisions, edge cases — outlives any developer, vendor, or stack.

1x → 10x

We turn your engineers into 10x engineers.

AI has commoditized coding speed — judgment is what's scarce. Codalio gives every engineer the leverage of a full team: they set direction, review, and decide, while the factory plans, builds, tests, and compliance-checks around them. One engineer, shipping at the scale of a team of twenty.

We augment your team. We don't replace it.

The organizations winning with agentic engineering aren't cutting their senior people — they're multiplying them. Codalio takes the judgment, domain knowledge, and standards already inside your team and gives them industrial leverage. Compliance in the blueprint. Brownfield-first. Granular, role-based permissions. Built for regulated industries — financial services, energy, critical infrastructure.

We bet on the orchestration era.

Every infrastructure wave ends the same way: what experts assemble by hand becomes a layer software generates. It happened to the cloud. It's happening to agentic coding — and Codalio was built for the second generation from day one.

CLOUD, THEN	Engineers hand-built every virtual machine, network, and storage volume — until orchestration turned "describe what you want" into automated infrastructure.
AI CODING, NOW	Senior engineers hand-configure sub-agents for every project — who plans, who tests, who reviews. Codalio is the orchestration layer: it designs that agentic flow itself.

GENERATION 1 — TODAY'S STATE OF THE ART

A senior engineer is the orchestrator

Senior engineer

→

reads requirements

→

writes tech spec

→

designs the agent team by hand

→

agent 1: code

agent 2: tests

agent 3: review

Every project needs a scarce expert to translate the business ask, architect the flow, and babysit the hand-offs. The expert is the bottleneck.

GENERATION 2 — CODALIO

The orchestration layer is the orchestrator

Business intent

→

Codalio orchestration layer

→

requirements

spec

architecture

agentic flow

→

engineer approves PR

The flow — hundreds of specialized agents with roles, hand-offs, and review gates — is generated per work item. Your experts supervise instead of assembling.

WHY MONOLITHIC PROMPTING BREAKS

As the codebase grows, the context grows past what any model holds reliably. One overloaded context window → drift, hallucinated requirements, different answers to the same prompt on every run.

HOW CODALIO KEEPS ACCURACY HIGH

architecture

data model

tests

debug

compliance

Each agent runs in a deliberately narrow context — one concern, one slice of the system. Shrinking the context per agent is precisely what makes agentic output accurate instead of lucky.

"Nobody was looking at the orchestration. Everybody was building sub-agents. We said: AI should design the flow itself — from the requirements to the spec to the architecture to the agents. That was the bet, and it's the moat."

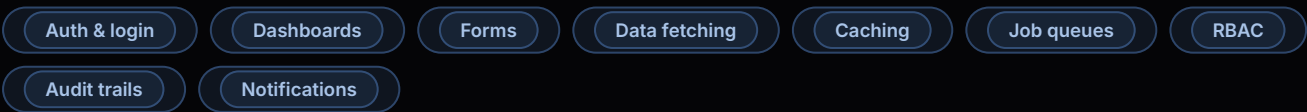
Don't regenerate what's already solved. Own what isn't.

Computer science first: most "new" enterprise code repeats patterns the industry has already perfected. Codalio splits the work at spec time — and keeps the spec and the code permanently in sync.



75-85% — commodity patterns. Instantiated from pre-validated components — deterministic, tested, identical on every run. No LLM roulette.

15-25% — your differentiation. AI-generated, then validated against your compliance rules and acceptance criteria.



The DNA loop — requirements are the new source code

Business requirements → PRD & spec → architecture → code

edge cases discovered at code level sync back into the spec

Coding always surfaces cases no one could see at the business level. Codalio writes them back into the product requirements — so the spec never drifts from reality. That living spec is your DNA: the complete, executable description of your system. Everything below it is generated.

Regenerate your entire system from the spec alone

Your DNA in a generic coding tool ~70-80%

Your DNA in Codalio ~98%

Stack migration, modernization, disaster recovery, vendor independence — you own the DNA, so you own the system. No more software held hostage by the three people who remember why it was built that way.

THE OLD WAY — A TRUE STORY FROM A MAJOR BANK

January — CEO: "Improve onboarding 25%" → 6 months interviewing engineers & scoping → \$20M plan lands on the CEO's desk → "Priorities have changed."

Intent → spec → validation against existing infrastructure → estimate: the most broken pipeline in enterprise software. Six months of senior time, zero shipped value.

THE SAME ASK, THROUGH CODALIO

8:00 AM — objective stated in plain English → 8:15 — PRD, spec & plan vs. existing architecture → Hours — build · test · debug · review · compliance → Same day — PR approved, running on staging

By the time the CEO has had coffee and driven to the office, somebody is already testing it. The decision happens while the question still matters.

Your roles evolve. We train them.

The hardest part of agentic engineering isn't the technology — it's the operating model. Codalio ships with the training and workshops that carry your organization through the change.

PRODUCT MANAGERS

Ticket writers



Specification authors — the spec is the new act of creation

ARCHITECTS

Diagram-keepers



Orchestration designers — mapping agent flows to enterprise standards

DEVELOPERS

Implementation typing



Validators & deciders — review, approve, and handle what machines can't

CTOS & LEADERS

Project shepherds



AI governors — owning spec frameworks, compliance gates, quality bars

Executive briefings

What agentic engineering changes for portfolio planning, budgeting, and governance.

Engineer enablement workshops

Hands-on: writing effective specs, supervising agentic flows, mastering review-and-approve.

Role-transition pathways

Structured guidance for PMs, architects, and engineering managers stepping into evolved roles.

First-initiative co-delivery

We run your first real initiative through the factory side-by-side with your team — the capability transfers, not just the tool.

One platform, the whole lifecycle

Spec-driven AI scoping

Technical specs — Jira & Linear-ready

PRD & spec versioning

Budget & timeline estimation

UI prototyping

Data model generation

Production-ready code — open-source Rhino foundation

Shared workspace

One-click DevOps & hosting



Start with one initiative.

Bring us the objective your team has been trying to scope for months. We'll show you the spec, the plan, and working software before your next steering committee meets.

codalio.com/enterprise

Sources: Garry Tan (Y Combinator), 2026 · Codalio Blog: From Vibe Coding to Agentic Engineering · MIT Project NANDA (2025) · CISQ (2022)