

EXECUTIVE AI LAB

Agentic AI for Enterprise Leaders

Move beyond AI experimentation.
Identify the work worth delegating. Build the system. Prove it.
Govern it. Scale it.

THE EXECUTIVE QUESTION

Where can AI agents safely take work off your organisation's critical path — and what evidence should you demand before they do?

A practical, business-led programme from discovery to deployment readiness.

01 / EXECUTIVE PERSPECTIVE

AI is moving from answers to action.

The next enterprise advantage is redesigning work around people, systems and controls.

FROM ASSISTANCE

Copilots and assistants help people produce, search, summarize and decide.

TO EXECUTION

Agents can coordinate multi-step work across documents, systems, rules and tools.

WITH CONTROL

Observability, evaluation, guardrails and human approval make deployment a leadership decision.

THE LEADERSHIP LENS

A strong Agentic AI programme answers six questions before technology becomes a priority.

1 Where is the value?

Which workflows have enough volume, friction and economic value to justify redesign?

2 What should be delegated?

Which decisions can be automated — and where should humans remain accountable?

3 What evidence is enough?

How will success, failure, cost, reliability and edge cases be measured?

4 What controls are required?

What data, permissions, escalation paths and guardrails are non-negotiable?

5 How does it reach operations?

What does the path from pilot to monitored production actually look like?

6 How does the organisation adopt it?

How do roles, governance and operating habits change when agents enter the workflow?

02 / PROGRAMME OUTCOMES

What leaders leave ready to do

The programme is designed around judgement, evidence and application — not passive exposure to AI concepts.

UNDERSTAND

Explain the difference between assistants, workflow automation and agents — and identify where each belongs.

MAP

Scan business functions for high-value agent opportunities and separate good candidates from poor ones.

DESIGN

Define goals, tools, memory, permissions, success criteria and guardrails before building.

BUILD

Create document-grounded, routing, planner-executor and function-specific agent flows.

TEST

Expose edge cases, silent failures, incorrect data, weak instructions and unsafe behaviour.

MEASURE

Read traces, latency, token use, cost, success rates and failure modes without needing to read code.

GOVERN

Translate acceptable use, human oversight, privacy, auditability and accountability into practical controls.

DEPLOY

Define a pilot-to-operations path with ownership, monitoring, permissions and change control.

“The objective is not to make leaders technical. It is to make them credible decision-makers about where AI should — and should not — enter the business.”

03 / OPERATING MODEL

From business problem to deployment evidence

Every lab follows a disciplined progression: start with the business brief, build against defined acceptance criteria, deliberately introduce ambiguity, inspect performance, then make a controlled decision.



THE EVIDENCE STACK

Acceptance criteria

What does "done" mean?

Trace & observability

What actually happened?

Failure-mode tests

Where does the system break?

Cost & performance

What does every run consume?

Guardrails

What is allowed, blocked or escalated?

Governance

Who owns the decision and the system?

THE STANDARD

Build less. Prove more.

A credible enterprise AI initiative should make its assumptions visible, expose its failure modes and show leaders what must be controlled before it scales.

- **Business value**

- **Technical fit**

- **Risk & controls**

- **Operational ownership**

DAY 01 / EXECUTIVE LAB

Agentic AI Foundations and the Opportunity Map

Turn Agentic AI from an abstract concept into a leadership decision framework. Learn what sits inside an agent, see an end-to-end agent in action and translate the technology into opportunities across your organisation.

1 What Agentic AI Is

Agents versus assistants and workflow automation; the goal-observe-reason-act-evaluate-repeat loop; simple reflex, model-based, goal-based, utility-based and learning agents; single- and multi-agent systems; autonomy and human-in-the-loop design.
Practical: Live demonstration: an agent completes a multi-step business task end to end.

2 Inside an Agent

Large language and reasoning models; tools and function calling; in-context, session and long-term memory; context windows and context engineering; retrieval and grounding; hallucination, non-determinism and output verification.
Practical: Leadership discussion: the questions to ask before trusting an agent.

3 The Agent Platform Landscape

The no-code, low-code and full-code spectrum; visual builders such as n8n, LangFlow and Flowise; enterprise platforms from Microsoft, Google and Azure; developer frameworks; LangSmith observability; build, buy or configure; integration, data residency, control, cost and exit criteria.
Practical: Lab: build an agent health and reporting dashboard in n8n.

4 Inside an Agent

Instruction anatomy — role, goal, context, tools, constraints and output; the C.L.E.A.R. framework; zero-, one-, few- and multi-shot prompting; reasoning prompts; structured outputs; tool descriptions; reusable instruction libraries and agent skills.
Practical: Hands-on: write and test agent instructions on participants' own work.

5 Mapping Opportunities Across the Business

High-value work in documentation, investigation, reconciliation, triage, reporting, analysis and correspondence; deterministic versus probabilistic processes; function-level scanning; value, feasibility and risk assessment; agent candidates and non-candidates.
Practical: Gamified task: spot agent opportunities across the functions in the room.

BY THE END OF THIS DAY

- Explain how agents work and how they differ from assistants and workflow automation.
- Evaluate platform choices against business and data requirements.
- Write stronger instructions using the C.L.E.A.R. framework and K-shot prompting.
- Create a first opportunity map for their business.

DAY 02 / EXECUTIVE LAB

Designing and Building Agents on a Visual Canvas

A guided build day in a browser-based, pre-configured environment. Assemble working agent patterns, inspect how information moves through the flow and make deliberate choices about confidence, escalation and safe behaviour.

6 The Visual Agent Canvas

GitHub Codespaces and the pre-configured lab; LangFlow models, prompts, agents, tools, file loaders and vector stores; flows, connections and hand-offs; credentials and global variables; running, inspecting and exporting a flow.
Practical: Lab: launch the environment and build a first working flow.

7 Building a Document-Grounded Agent

Retrieval-augmented generation; document loading and chunking; embeddings and vector stores; grounded answers and source citation; abstention when evidence is absent; testing against a policy document.
Practical: Lab: build a document question-answering agent with abstention.

8 Building a Triage and Routing Agent

Classification and intent detection; priority and severity scoring; routing rules and destinations; escalation paths; confidence levels and thresholds.
Practical: Lab: build an inbox triage and routing agent.

9 Multi-Agent Patterns

Planner-executor and critic-reviewer patterns; specialist agents with bounded roles; hand-off design and shared state; orchestration and failure containment; when a second agent adds value — and when it does not.
Practical: Lab: build a planner and executor hand-off.

10 Designing Guardrails Into the Flow

Input and output guards; sensitive-topic and out-of-scope handling; allowed and blocked actions; human approvals; read-only and write permissions; failure behaviour and safe defaults.
Practical: Individual and pair work: create an agent design brief covering goal, tools, memory, guardrails and success criteria.

BY THE END OF THIS DAY

- Build document-grounded, routing and multi-agent flows.
- Explain how information moves through an agent workflow.
- Make deliberate choices about confidence, escalation and safe behaviour.
- Create an agent design brief covering goal, tools, memory, guardrails and success criteria.

DAY 03 / EXECUTIVE LAB

Agents on Your Own Function

Put participants' own business functions at the centre of the learning. Each lab uses a realistic brief, supplied data, planted ambiguity and edge cases — ending with evidence against defined acceptance criteria.

11 The Function Labs

A controlled lab pattern with a business brief, isolated environment, planted ambiguity and verified end state. Participants select the case closest to their remit, choose a visual-canvas or coding-agent track, and define acceptance criteria before building.
Practical: Controlled discussion: select the right lab and define what a good result looks like.

12 Building With a Coding Agent

Coding agents and vibe coding with Claude Code; natural-language build instructions; reading and checking generated work; iterative improvement; and where a coding agent offers an advantage over a visual canvas.
Practical: Live demonstration and guided practice: build an automation by describing it.

13 The Function Lab Build

Work from the supplied brief and data; translate rules, policies and exceptions into the build; produce the required files and formats; iterate against fixed acceptance criteria.
Practical: Lab: build the selected function agent end to end.

14 Ambiguity, Edge Cases and Escalation

Find planted edge cases; handle missing, conflicting and out-of-policy data; design abstention and escalation behaviour; place human review points; identify silent failure in agent output.
Practical: Hands-on: surface and resolve the planted edge cases.

15 Proving the Agent Works

End-state verification; acceptance criteria; automated result checking; the definition of done; evidence for a reviewer; peer review of another participant's build.
Practical: Individual and pair work: run verification and peer-review another agent.

BY THE END OF THIS DAY

- Leave with a working function-specific agent.
- Record how the agent handled edge cases and escalation.
- Produce reviewer-ready evidence against defined acceptance criteria.
- Build against a real business brief rather than a toy example.

DAY 04 / EXECUTIVE LAB

Observability, Evaluation and Control

Make agent performance visible through traces, metrics, costs and failure-mode testing. Optimise against evidence and configure allow, block and escalate decisions that can be demonstrated in the trace.

16 Agent Observability

Traces, spans and runs; LangSmith projects and tracing; prompts, retrieval, tool calls, model responses, tokens, latency and cost; reading a trace without reading code.
Practical: Lab: instrument the function agent and inspect its traces.

17 Metrics, Baselines and Cost

Latency, token use, cost per run and success rate; cost drivers; runaway runs and loop detection; budgets, limits and cost controls; reporting performance to management.
Practical: Hands-on: record the metrics baseline for the function agent.

18 Failure Modes and Evaluation

Wrong tools, wrong data, fabricated results, silent failure, loops and refusal; failure-mode probes; test sets and expected results; evaluation-driven improvement; regression checking after change.
Practical: Lab: run a failure-mode probe and record what broke.

19 Optimisation

Improve instructions, retrieval settings, model choice, tool sets and task decomposition; compare before and after; manage quality, cost and latency trade-offs; apply model selection and routing.
Practical: Lab: complete an optimisation challenge measured trace to trace.

20 Guardrails and Controlled Decisions

The allow, block and escalate pattern; sensitive-term and out-of-scope controls; supported and unsupported topics; decision paths in the trace; proof that blocked requests reached neither model nor data.
Practical: Lab: configure guardrails and test every decision route.

BY THE END OF THIS DAY

- Present an agent's performance baseline.
- Explain its failure modes and record what broke.
- Demonstrate improvement trace to trace.
- Prove that guardrails behave as intended.

DAY 05 / EXECUTIVE LAB

Governance, Adoption and the Executive Decision

Put leaders in the decision seat: review an agent, challenge its evidence, define governance controls and determine the conditions for responsible deployment, adoption and scale.

21 The AI Review Board

Examine the reviewer's brief and agent output; test grounding, policy compliance and supporting evidence; review cost and reliability; define conditions, limits and review points; issue a go or no-go memo.

Practical: Case study: judge a deployed agent's work and make the executive decision.

22 Governance, Risk and Compliance

Build an AI governance toolkit covering acceptable use, human oversight, vendor due diligence, logging, audit trails, the agent register, change control, privacy, confidentiality, residency and accountability. Consider SDAIA principles and guidelines, ISO/IEC 42001, the EU AI Act and the NIST AI Risk Management Framework.

Practical: Controlled discussion: define governance guardrails for participants' organisations.

23 From Pilot to Operations

Define pilot scope and success criteria; ownership and support; integration, access and permissions; hosted, self-hosted and enterprise deployment options; production monitoring; change control and model updates.

Practical: Individual and pair work: create the pilot definition for the function agent.

24 Leading Adoption

Human and agent teams; role redesign and task reallocation; capability building; trust, transparency and resistance; measuring adoption and value; communicating agent use to customers and regulators.

Practical: Reflection: identify the leadership behaviours that decide adoption.

25 Emerging Methods

Context engineering; loop and graph engineering; agentic RAG; the Model Context Protocol; computer-use and browser agents.

Practical: Live demonstration: an agent uses external tools through MCP.

26 Capstone and Action Planning

Complete the capstone build or agent review; participate in the peer showcase; sequence the agent portfolio; create a function-level and personal ninety-day Agentic AI plan.

BY THE END OF THIS DAY

- Make a defensible go or no-go perspective.
- Translate governance into practical controls.
- Define a pilot-to-operations path.
- Align human adoption with measurable value.
- Create a practical ninety-day plan for responsible adoption.

05 / APPLIED BUSINESS CASES

Build around the work that matters

Participants can work on the function closest to their remit, using realistic friction rather than toy examples.

FINANCE

Invoice & purchase-order matching

Reconciliation, exception handling and controlled escalation.

HUMAN RESOURCES

Leave-request triage

Policy-grounded assessment with human review for exceptions.

SALES

Inbound lead triage & follow-up

Classification, prioritisation and response drafting.

MARKETING

Segmentation & campaign support

Structured analysis, content generation and review controls.

OPERATIONS

Order & shipment reconciliation

Cross-record comparison, discrepancy detection and routing.

CUSTOMER SUPPORT

Ticket triage & priority replies

Intent detection, severity scoring and escalation.

PROJECT MANAGEMENT

Project health & weekly reporting

Status synthesis, risk signals and management reporting.

BUSINESS ANALYSIS

Requirements & user stories

Document analysis, requirements extraction and structured outputs.

Executive principle: a use case is not “AI-ready” because it is technically possible. It is ready when value, ownership, controls, evidence and operational fit are clear.

06 / GOVERNANCE & CONTROL

Enterprise AI needs control by design

Governance is treated as part of the system — not a document created after the pilot.

DATA

Privacy, confidentiality, residency and grounding

ACCESS

Read/write permissions, credentials and least privilege

BEHAVIOUR

Allowed, blocked and escalated actions

HUMAN OVERSIGHT

Approval points, review thresholds and accountability

EVALUATION

Test sets, expected results, regression checks

OPERATIONS

Monitoring, ownership, change control and model updates

THE AI REVIEW BOARD

Before an agent becomes a production dependency, leaders examine:

- **Grounding & evidence**
- **Policy compliance**
- **Reliability & failure modes**
- **Cost & performance**
- **Human review points**
- **Security & permissions**

DECISION

GO / NO-GO / REDESIGN

07 / DELIVERY MODEL

Designed for senior decision-makers

The experience is built around executive discussion, live demonstrations, guided labs, case work, controlled failure testing and action planning.

WHO SHOULD ATTEND

- Chief Executives & Managing Directors
- CIO / CTO / CDO / Transformation leaders
- CFOs & Finance Directors
- COOs & Operations leaders
- Risk, Compliance & Internal Audit
- HR, Sales, Marketing & Business Excellence

DELIVERY EXPERIENCE

- Executive-led theory & case discussion
- Live agent demonstrations
- Pre-configured / guided build environments
- Individual & pair work
- Real-world friction and edge cases
- Reviewer-ready evidence & action planning

WHAT THE ORGANISATION WALKS AWAY WITH

1

An agent opportunity map

Prioritised opportunities across functions.

2

Working agent patterns

Function-specific builds grounded in real work.

3

Evidence of performance

Acceptance criteria, traces, metrics and failure findings.

4

Governance starting points

Controls for data, access, oversight and accountability.

5

Pilot definition

Scope, ownership, success criteria and operating requirements.

6

90-day action plan

A practical sequence from learning to responsible adoption.

THE OUTPUT IS NOT A CERTIFICATE. IT IS A DECISION-READY ORGANISATION.

Leaders leave with a clearer view of what to automate, what to test, what to control — and what to do next.

08 / ABOUT CEREBRAL HACKS

We teach the craft. Test it in competition. Deliver it for enterprise.

Cerebral Hacks brings together applied engineering, enterprise AI and high-intensity learning experiences — connecting capability building with real-world execution.

ENTERPRISE AI

Agentic systems, workflow redesign, AI adoption and deployment readiness.

APPLIED ENGINEERING

Cloud architecture, security, automation and production-minded technical practice.

BUILD & TEST

Hands-on labs, hackathons and controlled environments where ideas meet constraints.

OUR WORKING PRINCIPLE

1 DISCOVER

2 PRIORITIZE

3 DESIGN

4 BUILD

5 TEST

6 DEPLOY



LET'S IDENTIFY THE FIRST WORKFLOW WORTH REDESIGNING.

Cerebral Hacks

contact@cerebralhacks.com

cerebralhacks.com



SCAN TO VISIT