Chapter 4 Exercise Lab – Self-Learning User Guide

1. What this lab is for

This Exercise Lab is designed for **knowledge workers**, not engineers.

Chapter 4 is about **ethics, safety, and governance**. The exercises help you move from "I've read about these risks" to "I can spot and shape them in real situations," especially around:

- Over-trusting fluent AI answers
- Governance and accountability for real use cases
- Logging and the privacy paradox
- Alignment, fairness, and refusals
- Architecture choices for high-stakes scenarios (RAG, legal/medical, multi-agent)

You'll use simple interactive tools to **practice judgment, not coding**.

---

2. Getting started

1. **Open the lab**
   - Open Chapter4_Exercise_Lab.html in your browser (Chrome, Edge, etc.).
2. **Understand the layout**
   - **Left side:** Buttons for **Exercise 1–5**.
   - **Right side:** The selected exercise, with instructions, controls, and a reflection prompt.
3. **How to move around**
   - Click an exercise on the left to switch views.
   - Each exercise is self-contained; you can work through them in order (**1 → 5**) or jump to the one that fits a current project.

3. Exercise 1 – Trust Calibration Flight Deck

Theme: Illusion of confidence & human–AI trust

In the app:
Exercise 1 – Trust Calibration Flight Deck

What you'll practice

You'll practice **judging AI answers**, not just reading them.

- See multiple AI answers that all sound polished and confident.
- Decide how much you'd trust each answer and what you'd do with it.
- Discover which ones are actually **dangerous, incomplete, or responsible**.

You move from:

- "Fluent output = probably correct"
  to
- "I know where I must be skeptical, verify, or escalate."

How to use it

1. **Pick a scenario**
    - Use the **Scenario** dropdown:
        - Customer policy question
        - Internal financial summary
        - Health & wellness FAQ
2. **Read the three answers**
    - In **AI answers (A, B, C)**, click each answer card to expand it.
    - Read all three carefully. They all sound confident on purpose.
3. **Select an active answer**
    - Use the **Active answer** dropdown to choose **A**, **B**, or **C**.
    - Alternatively, click a card – it will become the active one.
4. **Set your trust level**
    - Move the **Trust level** slider from low to high.

    - The label shows a simple description (Very low / Low / Medium / High / Very high).

5. **Choose a planned action**
    - Select one of:
        - Send as-is
        - Light edit
        - Deep verification
        - Escalate to expert
6. **Save your decision**
    - Click **Save decision for this answer**.
    - Repeat for all three answers (A, B, and C).
7. **Reveal reality**
    - Once you've rated all answers, click **Reveal answer quality**.
    - The feedback explains:
        - Which answers are **dangerous**, **cautious**, or **responsible**.
        - Where you **over-trusted** or **under-trusted**.
        - How your chosen action (send, verify, escalate) compares to the risk.
8. **Reflect**
    - Use the reflection text at the bottom:
        - "Which answer would I have sent before this exercise?"
        - "In my real work, where am I at risk of over-trusting fluent but wrong answers?"

How to get value from it

- Apply this to **your real prompts**:
    - Imagine the AI answers are for your customers, leaders, or regulators.
    - Ask yourself: "Would I still send this as-is?"
- Use insights to create **simple internal rules**, for example:

    - "We never send health-related guidance without human review."
    - "We always verify numbers from AI against source systems."

4. Exercise 2 – Governance Triage Board

Theme: Risk, accountability, transparency

In the app:
Exercise 2 – Governance Triage Board

What you'll practice

You'll practice mapping **real AI uses** to the three pillars of governance:

- **Risk** – what could go wrong?
- **Accountability** – who owns it?
- **Transparency & Evidence** – how do we prove what happened?

You move from:

- "Governance is a checklist somewhere in Legal/Compliance"
  to
- "I can design a minimal governance plan for a specific AI use case."

How to use it

1. **Choose a scenario**
    - Use the **Scenario** dropdown:
        - AI drafting HR policies
        - Customer-facing pricing assistant
        - Internal legal research bot
2. **Review the governance cards**
    - In **Governance cards**, you'll see items such as:
        - Unlogged decisions flagged
        - Bias & unfair outcomes monitored
        - Named system owner

        - Risk committee review
        - Prompt & output logging
        - Source citations / RAG trace
        - Versioned model & config

3. **Cycle each card through the board**
    - Click a card to cycle it through:
        - Palette → **Risk** → **Accountability** → **Transparency & evidence** → back to palette.
    - As you click, the card “jumps” between the palette area and the three canvases.
4. **Build your governance layout**
    - For the chosen scenario, place cards in the columns where you think they belong.
    - Aim to have **at least one card** in each column.
5. **Check your governance plan**
    - Click **Check governance plan**.
    - The feedback explains:
        - Governance elements you placed well.
        - Important pieces you **missed** for this scenario (e.g., no named owner or no logging).
        - Whether you’re over-focusing on one pillar (e.g., lots of transparency, zero accountability).
6. **Reset if needed**
    - Click **Clear board** to return all cards to the palette and try a new scenario or new layout.
7. **Reflect**
    - Use the reflection prompt:

        - “Which cards would I insist on before approving a similar AI tool in my organization?”
        - “Who would I name as the owner, and how would we track risk over time?”

How to get value from it

- Treat this as a **rehearsal** for real design or vendor discussions.
- For each AI idea in your environment, ask:
    - “What’s the **Risk** column? The **Accountability** column? The **Transparency** column?”
- Capture your final layout as a **lightweight governance checklist** you can reuse.

5. Exercise 3 – Logging & Privacy Paradox Simulator

Theme: Auditability vs privacy and regulatory risk

In the app:
Exercise 3 – Logging & Privacy Paradox Simulator

What you’ll practice

You’ll practice tuning **logging settings** for different assistants and seeing the trade-offs between:

- Ability to reconstruct what happened (audit trail)
- Privacy and regulatory risk (PII, sensitive data, access)

You move from:

- “Logs are just an IT detail”
  to
- “I understand what we keep, why we keep it, and what makes it risky.”

How to use it

1. **Select an assistant type**
    - Use **Assistant type**:
        - Customer support bot

        - Internal HR assistant
        - Medical triage support

2. **Set logging options**
    - Toggle options such as:
        - Store full prompt text
        - Mask PII (names, emails, IDs)
        - Store only document IDs (no raw content)
        - Store full model outputs
        - Hash user ID instead of storing it directly
        - Restrict log access to compliance / security
3. **Recalculate scores**
    - Click **Recalculate scores**.
    - You’ll see two ratings:
        - **Audit & safety** (Low / Medium / High)
        - **Privacy & compliance risk** (Low / Medium / High)
    - The explanations describe:
        - How easy it would be to investigate incidents.
        - How much privacy/regulatory exposure you’re creating.
4. **View a sample log entry**
    - Scroll to **Sample log snippet**.
    - The log format changes to reflect:
        - Whether prompts are stored or masked.
        - Whether outputs and document IDs are logged.
        - Whether the user ID is hashed or raw.
5. **Try the balanced suggestion**
    - Click **Suggest balanced settings**.

   - The app picks a recommended combination for the scenario and updates the scores and snippet.
   - Compare the recommended setup to your initial instinct.

6. **Reflect**
   - Use the reflection question:
     - "If I had to defend these logging choices to a regulator, what would I say?"
     - "In my real environment, where are we over-logging or under-logging?"

How to get value from it

- Use this to **prepare questions** for your IT, security, or vendor teams:
  - "Do we store full prompts?"
  - "Is PII masked in logs?"
  - "Who has access to the logs?"
- Aim for a configuration that is:
  - **Auditable enough** to investigate incidents
  - **Not so invasive** that it creates unnecessary risk

6. Exercise 4 – Alignment & Fairness Testbed

Theme: Helpful, Honest, Harmless (HHH), fairness, refusals, determinism

In the app:
Exercise 4 – Alignment & Fairness Testbed

What you'll practice

You'll learn to think like an **evaluation designer**, not just an end-user:

- Create a small test set for a realistic scenario.
- See hypothetical model behaviors.
- Decide if each outcome is acceptable, needs mitigation, or unacceptable.
- Pick mitigations that match the problems.

You move from:

- “Alignment, bias, and refusals are abstract”
  to
- “I know how to test them in my own context.”

How to use it

1. **Choose a scenario**
   - Use **Scenario**:
     - Loan pre-qualification email
     - Internal promotion justification
     - Cybersecurity awareness message
2. **Choose an alignment focus**
   - Use **Alignment focus**:
     - Fairness across demographic variants
     - Excessive refusals vs legitimate requests
     - Deterministic vs varied outputs
3. **Select test cases**
   - In **Test cases**, check the boxes for the tests you want to include, such as:
     - Two applicants with identical profiles but different names
     - Same decision, but tone differs
     - Same input, different decisions on different runs
   - These are pre-built examples that mirror real equity and safety concerns.
4. **Run the simulation**
   - Click **Run simulation**.
   - For each selected test case, the app shows:
     - A short description of **simulated model behavior**.
     - A dropdown for your rating:

            - Acceptable
            - Needs mitigation
            - Unacceptable

5. **Rate each outcome**
    - For every test block, select the rating that matches your judgment.
    - Think about:
        - Would this be okay in your organization?
        - Would this raise a red flag with Legal, HR, or Risk?
6. **Select mitigations**
    - In **Mitigation choices**, click chips such as:
        - Add fairness constraints to prompt
        - Add human review for borderline cases
        - Lower temperature / narrow sampling
        - Expand and balance training examples
        - Log flagged decisions for audit
    - Active mitigations are highlighted.
7. **Evaluate your plan**
    - Click **Evaluate my plan**.
    - The feedback looks at:
        - How often your ratings match the intended alignment category.
        - Whether your mitigations align with the type of risk (fairness, refusals, determinism).
8. **Clear and iterate**
    - Use **Clear plan** to reset.
    - Try a different scenario or focus.

How to get value from it

- This mirrors what you might do before deploying an AI tool:
    - Design test cases
    - Decide what's acceptable
    - Define mitigations for bad behaviors
- Use your favorite tests from this exercise as **templates**:
    - "Three fairness checks we run on any hiring-adjacent tool."
    - "Two determinism checks for financial or safety-critical outputs."

7. Exercise 5 – High-Stakes Architecture Risk Lab

Theme: RAG privacy leaks, legal/medical workflows, multi-agent risks

In the app:
Exercise 5 – High-Stakes Architecture Risk Lab

What you'll practice

You'll practice designing and critiquing **simple AI architectures** using building blocks like:

- Vanilla LLM
- RAG over internal docs
- Sensitive index (HR, payroll)
- Safety filter
- Human-in-the-loop approval
- Legal reviewer, medical clinician
- Sales / Discount / Referee agents
- Role-based access control

You move from:

- "RAG, tools, and agents are buzzwords"
  to
- "I can spot obviously risky designs and propose safer ones."

How to use it

1. **Pick a high-stakes scenario**
   - Use **Scenario**:
     - Executive compensation Q&A bot
     - Internal legal research assistant
     - Patient education material generator
     - Multi-agent sales & discounting engine
2. **Study the baseline architecture**
   - Read the **Baseline** description at the top of the right card.
   - This baseline is intentionally flawed (e.g., RAG over all internal docs with no access control).
3. **Examine the current canvas**
   - In **Architecture canvas**, you'll see the blocks used in the baseline flow, such as:
     - RAG over internal docs
     - Vanilla LLM
     - Sales Agent / Discount Agent
4. **Add blocks from the palette**
   - On the left, **Architecture blocks** includes chips for:
     - Safety filter
     - Human-in-the-loop approval
     - Legal reviewer
     - Medical clinician
     - Sensitive index (payroll/HR)
     - Role-based access control
     - Referee Agent, etc.
   - Click a chip to add that block to the canvas.

- Each canvas block has a small “×” button to remove it.

5. **Design a safer architecture**
   - Add blocks that you think **reduce risk**, such as:
     - Role-based access control before a sensitive index
     - Human approval or legal review before final answers
     - Referee Agent for multi-agent setups
   - Remove blocks that make no sense for the scenario.
6. **Scan the architecture**
   - Click **Scan architecture**.
   - The **Risk heatmap** lists:
     - RAG privacy issues (e.g., RAG plus sensitive index without access control)
     - Missing human approvals in legal/medical contexts
     - Multi-agent collusion risk (e.g., Sales + Discount agents with no referee)
     - Lack of safety filtering
7. **Reset if needed**
   - Click **Reset to baseline** to restore the original flawed design and try again.
8. **Reflect**
   - Use the reflection prompt:
     - “If I had to explain this design to a regulator or risk committee, what safeguards would I highlight?”
     - “Which blocks are non-negotiable for this scenario?”

How to get value from it

- This is a **low-code way** to think like a solution architect:
    - You're not wiring systems; you're thinking about **paths and guardrails**.
- Use the blocks from this exercise as a **vocabulary** in internal conversations:
    - "Where does our RAG index point?"
    - "Do we have a human approval step for medical outputs?"
    - "Who is the referee when multiple agents can change discounts?"

8. Putting Chapter 4 into practice

As you complete the Chapter 4 Exercise Lab:

1. **Capture your decisions**
    - For each exercise, jot down:
        - One lesson about **trust**
        - One lesson about **governance**
        - One lesson about **logging**
        - One lesson about **alignment/fairness**
        - One lesson about **architecture**
2. **Connect the dots**
    - How does:
        - Over-trust in **Exercise 1** interact with poor logging in **Exercise 3**?
        - Governance gaps in **Exercise 2** show up as **architecture problems** in **Exercise 5**?
3. **Create simple guardrails for your own work**
    - Turn your insights into 3–5 practical rules, such as:
        - "For anything legal, medical, or financial, we always check sources or get human approval."

- “Every AI idea gets a governance triage: Risk, Accountability, Transparency.”
- “Logs must be just enough for audit, not a dump of raw PII.”
- “Before deployment, we run at least one fairness or refusal test from Exercise 4.”

Used this way, Chapter 4 stops being just a cautionary chapter and becomes a **playbook** for how you, personally, will work with AI in a safe, aligned, and defensible way.