

Exercise Lab – Self-Learning User Guide

1. What this lab is for

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

You'll use simple interactive controls (sliders, dropdowns, tables) to practice **real decisions about AI**:

- How retrieval settings affect answers (RAG)
- How to diagnose AI failures
- How to design safe tool use for agents
- How to structure multi-agent workflows
- How to balance speed, freshness, and safety in production

You don't need coding skills. You just need your **day-to-day work experience** and a willingness to experiment.

2. Getting started

1. Open the HTML file

- Double-click Chapter3_Exercise_Lab.html, or
- Drag it into a browser window (Chrome, Edge, etc.).


2. Understand the layout

- **Left side:** Navigation buttons for **Exercise 1–5**.
- **Right side:** The current exercise content (cards, sliders, tables, reflection boxes).

3. How to move around

- Click a button on the left (e.g., “Exercise 1 – RAG Control Panel”) to switch exercises.
- Only one exercise is visible at a time.

You can work through the exercises **in order** or pick the ones that match your current work challenges.

**Chapter 3 Exercise Lab**
Using AI safely and wisely at work

Exercise 1 – RAG Control Panel
Tune chunking & search

Exercise 2 – RAG Incident Room
Diagnose failures

Exercise 3 – Agent Toolbelt
Design safe tool use

Exercise 4 – Agent Storyboard
Map the workflow

Exercise 5 – Safety & Ops Room
Dial speed vs safety

Each exercise is written for knowledge workers, not engineers. Use it to practice real decisions with AI.



Exercise 1 – RAG Control Panel

In the app:

Exercise 1 – RAG Control Panel

Exercise 1 – RAG Control Panel

Objective: Tune retrieval settings for real work scenarios

Adjust "dials" for chunk size, hybrid search, and document count, then see how they affect coverage vs. noise for different business scenarios.

Step 1 – Choose a scenario & set the dials

Scenario

HR policy Q&A

Answer employee questions based on internal HR policies and handbooks.

Chunk size

Medium chunks – balanced context and precision

Hybrid search (keyword vs semantic)

About 50% semantic similarity and 50% keyword search.

Number of documents

5 document(s) will be pulled into the context window.

Run configuration

Step 2 – See coverage vs. noise

Context coverage (Recall)

Estimated recall: 75/100 (higher means you are less likely to miss relevant content).

Noise / Irrelevance (Precision loss)

Estimated noise: 57/100 (higher means more irrelevant or distracting content).

For HR policies, you typically want enough recall to avoid missing important rules, but not so much noise that outdated or unrelated clauses creep in. For HR, missing a clause can be risky. You usually want medium chunks and a small set of very relevant documents.

Reflection

In your own role, which configuration would you choose for this scenario and why?

What you'll practice

You'll build intuition for how RAG (Retrieval-Augmented Generation) behaves when you change:

- **Chunk size** (small vs large pieces of documents)
- **Hybrid search** (keyword vs semantic similarity)
- **Number of documents** retrieved

You'll see how these settings change:

- **Context coverage (Recall)**
- **Noise / Irrelevance (Precision loss)**

How to use it

1. **Choose a scenario**
 - Use the **Scenario** dropdown:

- HR policy Q&A
- Product SKU lookup
- Leadership summary
- Read the short description below the dropdown.

2. Adjust the “dials”

- Move the **Chunk size** slider (Small / Medium / Large).
- Move the **Hybrid search** slider (0–100% semantic).
- Move the **Number of documents** slider (3–10).

3. Run the configuration

- Click **Run configuration**.
- Watch:
 - The **Recall** gauge: how likely the system is to include the right content.
 - The **Noise** gauge: how much irrelevant material may show up.

4. Read the explanation

- The text below the gauges explains why those settings might be good or risky for the chosen scenario.

5. Write your reflection

- In the **Reflection** box, answer:

“In your own role, which configuration would you choose for this scenario and why?”

How to get value from it

- Try **extremes** first (tiny chunks vs huge chunks; very few vs many documents).
 - Ask yourself: *“If this were my AI assistant at work, would I trust its answers under this configuration?”*
 - Capture one or two “rules of thumb” you discovered (e.g., “For HR questions, avoid too many docs—noise goes up fast.”).
-

Exercise 2 – RAG Incident Room

In the app:

Exercise 2 – RAG Incident Room

Exercise 2 – RAG Incident RoomObjective: Diagnose what broke in a RAG incident

Read short "incident" stories and decide which part of the RAG triad failed. Practice thinking like a product owner for an AI system.

Step 1 – Choose an incident

Incident

Incident A – Plausible but wrong

The system answered a HR eligibility question with high confidence, but the answer turned out to be wrong.

Question: "Am I eligible for paid parental leave if I have been with the company for 8 months?"

Step 2 – Review answer & context

AI answer

Yes, you are eligible for 12 weeks of paid parental leave. Our policy states that any employee with more than 6 months of tenure qualifies for full parental leave.

Retrieved snippets

- Excerpt from Parental Leave Policy (updated 2020): Employees with 12+ months of continuous service are eligible for paid parental leave.
- Benefits FAQ: 'Paid parental leave requires one year of service at the time of the event.'

Latency breakdown

Retrieval: 0.9s | Model: 0.8s | Tools: 0.2s

Step 3 – Diagnose what broke

Part of the RAG triad	Status
Context relevance (retrieval)	OK
Grounding / faithfulness	Broken
Answer relevance (to user question)	OK

Where is latency coming from?

Mostly model / reasoning

Show diagnosis summary

You marked context as ok, grounding as broken, and answer relevance as ok. In Incident A, retrieval actually found the right policy excerpts, but the answer ignored them. That points to grounding/faithfulness being broken: the model is hallucinating over correct context. You also noted latency is mostly from model/reasoning. That gives you a hint about where to optimize performance in addition to quality.

Product owner note

If you owned this system, where would you ask your team to focus first?

What you'll practice

You'll practice **diagnosing AI failures** using the **RAG triad**:

- **Context relevance** – Did it retrieve the right documents?
- **Grounding / faithfulness** – Did the answer stay true to those documents?
- **Answer relevance** – Did it answer the user's actual question?

You'll also notice where **latency** (slowness) comes from.

How to use it

1. **Pick an incident**

- Use the **Incident** dropdown (A, B, or C).
- Read:
 - The description
 - The user's question
 - The AI answer
 - The retrieved snippets
 - The latency breakdown

2. Judge each part of the triad

- For each row in the table:
 - Context relevance (retrieval)
 - Grounding / faithfulness
 - Answer relevance
- Choose **OK** or **Broken**.

3. Identify latency source

- Use the **Where is latency coming from?** dropdown:
 - Mostly retrieval
 - Mostly model / reasoning
 - Mostly tools / APIs

4. Show the diagnosis summary

- Click **Show diagnosis summary**.
- Read the explanation: it will connect your choices to a likely root cause.

5. Optional: product owner note

- In **Product owner note**, write what you'd ask your AI team to fix first.

How to get value from it

- Don't think in terms of "the model is bad." Instead ask:
 - Did it **find** the right content?

- Did it **stick** to that content?
 - Did it **answer** the question asked?
 - Think about your own environment:
 - *If an AI system made this mistake at work, how would you talk about it in a post-mortem?*
-

Exercise 3 – Agent Toolbelt Designer

In the app:

Exercise 3 – Agent Toolbelt Designer

Exercise 3 – Agent Toolbelt Designer

Objective: Decide which tools an agent should use first, optionally, or never

Treat the AI assistant like a digital colleague. Decide which tools it must use, may use, or should never use in specific business scenarios.

Step 1 – Choose a scenario

Scenario

Customer billing assistant

The assistant helps agents answer customer questions about invoices, payments, and credits.

Available tools

- Policy search – Looks up billing and credit policies.
- Billing system lookup – Fetches invoice and payment records.
- CRM lookup – Shows full customer history and notes.
- Web search – Searches the public web.
- Email sender – Sends emails on your behalf.

Step 2 – Assign tools to zones

Tool	Zone
Policy search	Must use first
Billing system lookup	Use if needed
CRM lookup	Use if needed
Web search	Use if needed
Email sender	Never use for this scenario

If a tool fails, the agent should...

Ask the user what to do

Review my toolbelt

You set 1 tool(s) to 'Must use first', 3 to 'Use if needed', and 1 to 'Never use'. Your zones are generally aligned with the idea that internal, authoritative systems should lead, with broader tools used more cautiously. With a failure strategy of 'Ask the user', you keep the human in control but add friction.

Reflection

In your real work, which tools or systems should never be used automatically by an AI assistant and why?

What you'll practice

You'll design a **toolbelt** for an AI assistant:

- Which tools it must use first
- Which tools it may use if needed
- Which tools it must never use in a given scenario
- How it should behave when a tool fails

You're expressing **policy and governance**, not coding.

How to use it

1. **Choose a scenario**

- Use the **Scenario** dropdown:
 - Customer billing assistant
 - Internal HR policy helper
 - Sales pipeline explainer
- Read the scenario description.

2. Review available tools

- Read the **Available tools** list (e.g., Policy search, CRM lookup, Web search, Email sender).
- Notice what each tool can do.

3. Assign tools to zones

- In the table:
 - For each tool, pick a **Zone**:
 - Must use first
 - Use if needed
 - Never use for this scenario
- Make sure you've chosen a zone for every tool.

4. Choose a failure strategy

- In **If a tool fails...**, pick:
 - Retry with backoff
 - Ask the user
 - Try a different tool
 - Fail fast

5. Review your toolbelt

- Click **Review my toolbelt**.
- Read the narrative summary and any safety hints.

6. Write your reflection

- In **Reflection**, answer:

“In your real work, which tools or systems should never be used automatically by an AI assistant and why?”

How to get value from it

- Think of **real systems** in your company:
 - Which ones are “source of truth”?
 - Which ones are risky if misused?
 - Use your answers as a starting point for an **AI governance checklist**:
 - “These tools can be auto-used; these require human confirmation.”
-

Exercise 4 – Agent Team Storyboard

In the app:

Exercise 4 – Agent Team Storyboard

Exercise 4 – Agent Team Storyboard

Objective: Design the flow between Research, Writer, and Verifier agents

Map out a simple multi-agent workflow. Decide which agent handles each step and in what order, then “run” it to see if anything important is missing.

Step 1 – Choose a workflow

Workflow

Competitor briefing for leadership

Prepare a concise competitor briefing for leadership, based on recent market information and internal notes.

Step 2 – Assign tasks to agents

Task	Agent lane	Order (1–6)
Gather relevant internal and external documents	Research agent	1
Extract key facts, risks, and trends	Writer agent	2
Draft a 1-page briefing	Writer agent	3
Check for any policy or disclosure issues	Verifier / Guardian agent	4
Polish tone and clarity for leadership	Writer agent	5

Step 3 – Run your storyboard

Run my workflow

You included at least one Verifier/Guardian step at order 4. Consider moving it closer to the final output so it can review the full result. You allow Research to inform Writing, which reduces hallucination risk and improves grounding. Overall, think of this storyboard as a design for both human and AI roles. Where you place the Verifier is especially important for high-risk content.

Reflection

What is one change you would make to your real-world process to mirror this best-practice chain?

What you’ll practice

You’ll design a **multi-agent workflow**:

- Which tasks belong to a **Research agent**
- Which tasks belong to a **Writer agent**
- Which tasks are handled by a **Verifier / Guardian agent**
- In what **order** these tasks should run

This mirrors how you might structure both **human** and **AI** roles.

How to use it

1. Choose a workflow

- Use the **Workflow** dropdown:
 - Competitor briefing for leadership
 - Customer complaint case summary

- HR policy announcement
- Read the description.

2. Assign tasks to agents

- In the table:
 - Each row is a task (e.g., Gather docs, Draft summary, Check compliance).
 - For each task:
 - Choose an **Agent lane**:
 - Research agent
 - Writer agent
 - Verifier / Guardian agent
 - Set an **Order** number (1–6) to show the sequence.

3. Run your workflow

- Click **Run my workflow**.
- Read the feedback:
 - Is there a Verifier step?
 - Is the Verifier near the end?
 - Does writing start before any research?

4. Adjust if needed

- If the feedback shows problems (e.g., Writer before Research), adjust agents or order and run again.

5. Write your reflection

- In **Reflection**, answer:

“What is one change you would make to your real-world process to mirror this best-practice chain?”

How to get value from it

- Compare this agent storyboard to how your **human team** works today.

- Ask yourself:
 - *“Do we actually have a Verifier role before big decisions or announcements?”*
 - *“If we add AI, where should that Verifier step live?”*
-

Exercise 5 – Safety & Ops Control Room

In the app:

Exercise 5 – Safety & Ops Control Room

Exercise 5 – Safety & Ops Control RoomObjective: Design Experiment vs Production operating modes

Configure a "Lab" mode and a "Production" mode for your AI assistant. Balance speed, freshness, and safety for each environment.

Step 1 – Configure Experiment (Lab) mode

Guardian agent

Off

Citation verifier

On

Allowed sources

Internal only

Caching

None

Monitoring

Basic logs only

Step 2 – Configure Production mode

Guardian agent

On

Citation verifier

On

Allowed sources

Internal only

Caching

Moderate

Monitoring

Basic logs only

Step 3 – Compare speed, freshness, and safety

Experiment mode gauges

SpeedSpeed: 45/100

FreshnessFreshness: 70/100

SafetySafety: 80/100

Production mode gauges

SpeedSpeed: 55/100

FreshnessFreshness: 60/100

SafetySafety: 100/100

Compare Lab vs Production

In Experiment (Lab) mode, your configuration yields Speed 45/100, Freshness 70/100, and Safety 80/100. In Production mode, you have Speed 55/100, Freshness 60/100, and Safety 100/100. Compare these trade-offs with your risk appetite: high-risk content usually deserves higher safety even if it costs speed.

Leadership message

Write one sentence you would use with leadership to justify your Production mode choices.

What you'll practice

You'll configure two modes for an AI system:

- **Experiment (Lab) mode** – where you explore and test
- **Production mode** – where real users rely on the system

You'll see how choices impact:

- **Speed**
- **Freshness of information**
- **Safety & compliance**

How to use it

1. Configure Lab mode

- In the first card, set:
 - Guardian agent: On/Off
 - Citation verifier: On/Off
 - Allowed sources: Internal only / Internal + web
 - Caching: None / Moderate / Aggressive
 - Monitoring: Basic logs / Dashboards + alerts

2. Configure Production mode

- In the second card, set the same controls, but:
 - Imagine this is what you'd put in front of **real employees or customers**.
 - Aim for the configuration you'd be comfortable defending.

3. Compare modes

- Click **Compare Lab vs Production**.
- Look at the gauges:
 - Speed (0–100)
 - Freshness (0–100)

- Safety (0–100)

- Read the text summary of your trade-offs.

4. Write your leadership message

- In **Leadership message**, write:

“One sentence you would use to justify your Production mode choices to leadership.”

How to get value from it

- Ask:
 - *“Is my Production mode clearly safer than Lab?”*
 - *“Where am I willing to trade speed for safety?”*
 - Use your leadership message as a draft for an **AI usage standard** in your team or department.
-

8. Putting it all together

As you finish the Chapter 3 Exercise Lab:

1. Review your reflection answers across the exercises.
2. Collect them into a short **“AI Usage Playbook”** for yourself:
 - How you’d configure RAG for your work
 - How you’d diagnose failures
 - How you’d control tool use and workflows
 - How you’d set Lab vs Production modes

This lab is not about learning jargon. It is about becoming a **confident, thoughtful user and governor of AI systems** in your real job.

If you revisit these exercises after a few weeks of using AI at work, you’ll often make **different choices**—and that’s a sign your understanding has deepened.