AI Agents 101: Facilitator Guide

AI Agents 101: Facilitator Guide

The views expressed are the author's own and do not necessarily represent the views of the U.S. Census Bureau or the U.S. Department of Commerce.

This guide helps you deliver the course or adapt it for your team. The materials are designed to work as-is, but you know your audience better than we do.

Overview

Duration: 40-60 minutes (flexible)

Format: Presentation + live demo + discussion

Core message: Bounded agency beats autonomous agents. Start simple, add complexity only when justified.

Materials Checklist

File	Purpose	When to Use
`slides.md` / `slides.pdf`	Visual scaffolding for presentation	During delivery
`student_handout.md`	Quick reference for attendees	Distribute before or after
`exercises.md`	Post-session practice	Share after session
`course_notes.md`	Full speaker notes and depth	Your prep; reference for Q&A

Recommended prep reading: Microsoft AI Red Team, "Taxonomy of Failure Modes in Agentic AI Systems" (2025). Understanding what goes wrong helps you explain why the design principles matter.

Timing Guide (40-minute version)

Section	Time	Slides	Notes
Opening + Problem	5 min	1-3	Set the stakes. Salesforce story lands well.
Vocabulary	8 min	4-7	Don't rush. These terms do the heavy lifting.
Live Demo	12 min	8-10	Run the prompt live. Let them see it work.
Case Study	5 min	11-12	Gloss over details. Point is "bounded agency at scale."
Design Principles + Stakes	8 min	13-17	Hit 2-3 principles hard, use Stakes slide to land "why it matters."
Close + Q&A	5 min	18-20	Point to resources. Take questions.

If running short: Cut the case study to one sentence ("Here's what this looks like at scale — 7,000 items, $15, complete audit trail"). Spend the time on vocabulary and demo instead.

If running long: The demo and Q&A are where time expands. Set a hard stop for the demo; you can always say "try breaking it yourself later."

Timing Guide (60-minute version)

Section	Time	Slides	Notes
Opening + Problem	5 min	1-3	Same as above
Vocabulary	10 min	4-7	Take questions here. Clarify terms.
Live Demo	15 min	8-10	Run prompt, then try one "break it" scenario live
Case Study	10 min	11-12	Walk through the architecture briefly
Design Principles + Stakes	12 min	13-17	Cover all six with examples; Stakes slide connects to real failures
Close + Q&A	8 min	18-20	Extended discussion

Live Demo Tips

Before the session: - Have the recipe prompt ready to paste (it's in `exercises.md` and `student_handout.md`) - Test it once to make sure your chat interface is working - Have a backup screenshot in case internet/tools fail

During the demo: - Narrate what's happening: "Watch — it's selecting a recipe and explaining why" - Point out the hard stop when it hits the allergen check - Show the iteration loop ("anything you already have?")

If it does something unexpected: That's a teaching moment. Ask the audience: "What would you change in the prompt to prevent this?"

Common Questions (and Short Answers)

"How is this different from just using ChatGPT?" > It's not magic — it's structure. The prompt creates a workflow with decision points, constraints, and checkpoints. You could do this yourself, but the structure makes it repeatable and auditable.

"Do I need to code to do this?" > No. The recipe prompt runs in any chat interface. The skill is specification, not programming.

"What about hallucinations?" > Bounded agency helps. Explicit criteria, required reasoning, human checkpoints — these reduce (but don't eliminate) the risk. Design for uncertainty.

"When should I NOT use an agent?" > When a simple prompt works. When a human decision takes 30 seconds. When the stakes are high and you can't verify outputs. Most problems don't need agents.

"What tools do you recommend?" > Start with what you have access to. Chat interfaces (Claude, ChatGPT) are fine for learning. Don't buy tools until you've outgrown the free tier.

"What are the biggest risks with AI agents?" > Microsoft's AI Red Team catalogued them: agents acting outside their intended scope, cascading failures when there's no stop condition, misalignment between what you asked for and what it does, denial of service from runaway loops. The design principles directly address these — that's why they matter.

"What's a circuit breaker?" > It's a fail-safe — a mechanism that stops the agent when predefined conditions are met. Token limits, iteration caps, confidence thresholds. The concept comes from engineering: electrical circuit breakers, dead man's switches. An agent that can't stop itself is an agent you can't trust.

Adapting for Your Audience

More technical audience: - Spend less time on vocabulary - Go deeper on the case study architecture - Discuss confidence thresholds and routing logic

Less technical audience: - More time on vocabulary, more repetition - Keep the demo simple — don't try to break it live - Focus on the "reviewing agent behavior" checklist

Leadership/governance audience: - Lead with the Salesforce story - Emphasize "autonomy is a governance choice" - Focus on the evaluation questions: "Would I sign my name under this?"

Skeptical audience: - Acknowledge the hype upfront — you're on their side - Emphasize what agents CAN'T do well - The "most problems don't need agents" principle lands well here

---

After the Session

- Share the student handout and exercises
- Point them to the course notes if they want depth
- Invite them to try the exercises and report back what they learned

---

License

Course: AI Agents 101 — Bounded Agency Over Autonomous Agents