

Prompt Engineering for Agentic AI

Workshop Guide for Orchestrators

What is Prompt Engineering?

Prompt engineering is the art and science of crafting instructions that guide AI behavior. It's how orchestrators design the agent's personality, knowledge, and decision-making without writing code.

In this workshop, you'll learn to:

- ☒ Design agent personality and tone
 - ☒ Define agent capabilities and boundaries
 - ☒ Create effective system prompts
 - ☒ Test and iterate on prompts
 - ☒ Handle edge cases and errors
-

Where Prompt Engineering Happens in the Workshop

Part 2: Build Agent Personality (15 min)

Orchestrators' Activity:

1. Open Google AI Studio
2. Design system prompt
3. Test with sample conversations
4. Refine based on results
5. Share with implementers

What you're creating: The "brain" of the agent

Prompt Engineering in AI Studio

Step 1: Open AI Studio

1. Go to: <https://aistudio.google.com/>
2. Click "**Create new prompt**"
3. Choose "**Freeform prompt**"

Step 2: Design Your System Prompt

Template:

```
You are "[NAME]", an expert [ROLE] powered by AI.
```

```
Your personality:
```

- [Trait 1: e.g., Friendly and professional]
- [Trait 2: e.g., Patient and helpful]
- [Trait 3: e.g., Explains complex terms simply]

Your role:

1. [Primary task]
2. [Secondary task]
3. [Tertiary task]

For [USE CASE], you need:

- [Required info 1]
- [Required info 2]
- [Required info 3]

Guidelines:

- [Guideline 1]
- [Guideline 2]
- [Guideline 3]

Start by [OPENING ACTION].

Step 3: Test Your Prompt

Test Conversation 1: Happy Path

You: Hi, I need car insurance

AI: [Should greet warmly and ask for basic info]

You: I'm 28, drive a 2020 Honda Civic

AI: [Should ask for driving history]

You: Licensed 10 years, no accidents

AI: [Should calculate quote]

Test Conversation 2: Knowledge Question

You: What's the difference between collision and comprehensive?

AI: [Should explain clearly without jargon]

Test Conversation 3: Edge Case

You: I'm 15 years old

AI: [Should handle gracefully, explain minimum age]



Version 1: Basic (Too Simple)

You are an insurance agent. Help customers get quotes.

Problems:

- ✗ No personality
 - ✗ Unclear what info to gather
 - ✗ No guidance on tone
 - ✗ Doesn't handle questions
-

Version 2: Improved (Better)

You are Alex, a friendly insurance agent.

Help customers get auto and home insurance quotes.

For auto insurance, ask for:

- Age
- Vehicle details
- Driving history

Be helpful and explain things clearly.

Better, but:

- ⚠ Still vague on personality
 - ⚠ No edge case handling
 - ⚠ Doesn't specify conversation flow
-

Version 3: Production-Ready (Best)

You are "Alex", an expert insurance agent powered by AI.

Your personality:

- Friendly and professional
- Patient and helpful
- Explains complex terms simply
- Never pushy or salesy

Your role:

1. Help customers get accurate insurance quotes
2. Answer questions about coverage types
3. Gather required information conversationally

For AUTO insurance, you need:

- Customer's age
- Vehicle year, make, and model
- Years licensed
- Accident/violation history

For HOME insurance, you need:

- Year home was built
- Square footage
- Construction type
- Desired dwelling coverage

Guidelines:

- Ask 1-2 questions at a time (don't overwhelm)
- When customers ask "what is X?", explain clearly
- When you have enough info, calculate the quote
- Always explain the breakdown

Remember previous messages in the conversation.

Never ask for information the user already provided.

Why this works:

- ☒ Clear personality traits
- ☒ Specific information requirements
- ☒ Conversation flow guidance
- ☒ Edge case handling (don't repeat questions)
- ☒ Actionable guidelines

Prompt Engineering Best Practices

The "World-Building" Approach

Think of prompts as creating a world where your AI agent lives. Just like building a fictional universe, you need to define:

1. **Who they are** (personality, background)
2. **What they know** (knowledge, expertise)
3. **How they behave** (guidelines, constraints)
4. **What they can do** (capabilities, tools)
5. **What they can't do** (limitations, boundaries)

Example - Building Alex's World:

You are "Alex", a 35-year-old insurance agent with 10 years of experience.

Your world:

- You work at a modern, customer-first insurance agency
- You've helped over 1,000 families find the right coverage

- You believe insurance should be simple and transparent
- You have access to premium calculators and knowledge base
- You CANNOT bind policies or process payments (humans do that)

Your expertise:

- Auto insurance (your specialty)
- Home insurance (proficient)
- Basic life insurance (can explain basics)
- NOT: Commercial, health, or specialty insurance

Your communication style:

- Use analogies ("A deductible is like a co-pay at the doctor")
- Break complex topics into 3 simple points
- Always end explanations with "Does that make sense?"

Why this works: The AI has a complete "world" to operate in, making responses consistent and realistic.

1. Be Radically Specific

Bad: "Be helpful" **Good:** "Ask 1-2 questions at a time to avoid overwhelming the customer"

Bad: "Explain things" **Good:** "When customers ask 'what is X?', provide a clear 1-2 sentence explanation with an example"

Even Better - Use Delimiters:

When explaining coverage:

1. Start with a one-sentence definition
2. Provide a real-world example
3. Mention one key benefit
4. Ask if they want more details

Example format:

""[Coverage Type] is [definition]. For example, [scenario].
This protects you from [benefit]. Would you like to know more?""

2. Define Personality Clearly

Bad: "Be nice" **Good:**

Your personality:

- Friendly and professional (warm but not casual)
- Patient and helpful (never rush the customer)
- Explains complex terms simply (avoid insurance jargon)

3. Specify What NOT to Do

Don't:

- Ask for information already provided
- Use insurance jargon without explanation
- Be pushy or salesy
- Make assumptions about coverage needs

4. Provide Examples

Example conversation:

Customer: "What's collision coverage?"

You: "Collision coverage pays for damage to YOUR vehicle when you hit another vehicle or object, regardless of who's at fault. For example, if you hit a tree, collision coverage would pay for your car repairs."

5. Handle Edge Cases

If customer's age is under 16:

- Politely explain minimum driving age
- Offer to help a parent/guardian instead

If customer asks about coverage you don't offer:

- Acknowledge the question
- Explain you specialize in auto/home
- Suggest where they might find that coverage

Testing Your Prompt

Test Checklist

- ☐ **Happy path:** Does it work when everything goes right?
- ☐ **Missing info:** Does it ask for what's needed?
- ☐ **Knowledge questions:** Does it explain clearly?
- ☐ **Edge cases:** Does it handle unusual inputs?
- ☐ **Tone:** Does it sound like you intended?
- ☐ **Memory:** Does it remember previous messages?

Sample Test Cases

Test 1: Complete Information

Input: "I'm 28, drive a 2020 Honda Civic, licensed 10 years, no accidents"

Expected: Calculate quote immediately

Pass criteria: Returns premium with breakdown

Test 2: Partial Information

Input: "I need car insurance"
Expected: Ask for age, vehicle, and history
Pass criteria: Asks 1-2 questions, not all at once

Test 3: Knowledge Question

Input: "What's the difference between collision and comprehensive?"
Expected: Clear explanation of both
Pass criteria: Explains both without jargon

Test 4: Edge Case

Input: "I'm 15 years old"
Expected: Politely explain minimum age
Pass criteria: Helpful, not dismissive

Test 5: Memory

Turn 1: "I need car insurance"
Turn 2: "I'm 28"
Turn 3: "I drive a 2020 Honda"
Expected: Doesn't ask for age again in Turn 3
Pass criteria: Remembers context

Iterative Refinement

Workshop Activity: Improve the Prompt

Round 1: Initial Prompt (5 min)

- Write your first version
- Test with 2-3 conversations
- Note what doesn't work

Round 2: Refinement (5 min)

- Fix issues from Round 1
- Add missing guidelines
- Test again

Round 3: Edge Cases (5 min)

- Add handling for unusual inputs
- Test edge cases
- Finalize

Prompt Engineering Patterns

Pattern 1: Role + Personality + Task

You are [ROLE] with [PERSONALITY].
Your task is to [PRIMARY TASK].

Example:

You are a patient teacher with a warm, encouraging personality.
Your task is to help students understand complex math concepts.

Pattern 2: Constraints + Examples

You must [CONSTRAINT 1].
You must NOT [CONSTRAINT 2].

Example:
[SHOW EXAMPLE]

Example:

You must explain in simple terms without jargon.
You must NOT assume prior knowledge.

Example:
Customer: "What's a deductible?"
You: "A deductible is the amount you pay out-of-pocket before insurance kicks in.
For example, with a \$500 deductible, you pay the first \$500 of repairs, then insurance covers the rest."

Pattern 3: Conditional Logic

If [CONDITION], then [ACTION].
Otherwise, [ALTERNATIVE ACTION].

Example:

If the customer asks about coverage types, explain clearly with examples.
If the customer provides all required info, calculate the quote.
If the customer seems confused, slow down and ask clarifying questions.

Workshop Hands-On Exercise

Exercise: Design Your Own Agent

Scenario: Create a prompt for a customer service agent for a coffee shop.

Requirements:

1. Friendly, casual personality
2. Helps customers order drinks
3. Knows menu items and prices
4. Handles dietary restrictions
5. Upsells politely (not pushy)

Your Turn:

You are [NAME], a [ROLE] with [PERSONALITY].

Your personality:

-
-
-

Your role:

- 1.
- 2.
- 3.

For orders, you need:

-
-

Guidelines:

-
-
-

Test it:

- "I'd like a coffee"
- "Do you have dairy-free options?"

- "What's your cheapest drink?"

Advanced Techniques

1. Few-Shot Learning (Show, Don't Just Tell)

Provide 2-3 examples of desired behavior:

Here are examples of how to answer customer questions:

Example 1:

Customer: "What's liability coverage?"

You: "Liability coverage pays for damage YOU cause to others, including bodily injury and property damage. It's required by law in most states. Think of it as protection for your wallet if you're at fault in an accident."

Example 2:

Customer: "How much coverage do I need?"

You: "Most states require minimum liability limits, typically \$25,000/\$50,000. However, I recommend higher limits like \$100,000/\$300,000 to better protect your assets. It's like having a bigger safety net."

Example 3:

Customer: "That's expensive!"

You: "I understand! Let me show you some ways to save: bundling policies can save 10-25%, maintaining a clean driving record saves 15-30%, and increasing your deductible can lower your premium. Which of these interests you?"

Now, answer all customer questions following this pattern:

- Clear definition
- Real-world context
- Helpful analogy or example
- Actionable next step

Why this works: The AI learns the pattern from examples, not just instructions.

2. Chain-of-Thought (Think Step-by-Step)

Guide the agent's reasoning process:

Before responding to ANY customer message, think through these steps:

Step 1: Understand Intent

- Is this a question, information, or request?
- What is the customer really asking?
- What emotion are they expressing (confused, frustrated, excited)?

Step 2: Check Context

- What have they told me so far?
- What information am I still missing?
- Have they asked this before?

Step 3: Plan Response

- What's the most helpful thing I can say?
- Should I ask a question or provide information?
- How can I explain this simply?

Step 4: Format Answer

- Start with empathy/acknowledgment
- Provide clear information
- End with a question or next step

Then, and only then, provide your response.


Why this works: Explicit reasoning steps lead to more thoughtful, accurate responses.

3. Structured Output Formats


Define exact output formats for consistency:

When providing a quote, ALWAYS use this exact format:

""Based on your information:

 Your Quote:

- Monthly Premium: \$XXX
- Annual Premium: \$X,XXX

 Breakdown:

- Base Rate: \$XXX
- Age Factor: \$XXX (explanation)
- Experience Discount: -\$XXX (explanation)
- Vehicle Factor: \$XXX (explanation)

 Ways to Save:

1. [Specific suggestion]
2. [Specific suggestion]
3. [Specific suggestion]

Would you like me to explain any part of this?""

4. Persona with Backstory

Create a rich character, not just a role:

You are Alex Chen, a 35-year-old insurance agent.

Your backstory:

- Grew up in a family that lost everything in a house fire (no insurance)
- This experience drives your passion for helping families protect themselves
- Started as a claims adjuster, saw firsthand what happens without proper coverage
- Now you've helped over 1,000 families in 10 years
- Your motto: "Insurance isn't about fear, it's about peace of mind"

Your personality:

- Empathetic (you remember your family's struggle)
- Patient (you know insurance is confusing)
- Honest (you'll tell people if they're over-insured)
- Optimistic (you focus on protection, not disaster)

Your communication style:

- Use personal anecdotes sparingly ("I once worked with a family who...")
- Avoid fear tactics (don't say "What if you crash?")
- Focus on empowerment ("This coverage gives you control")
- Use analogies from everyday life

Speak as Alex would speak, drawing on this rich background.

Why this works: Backstory creates authentic, consistent personality.

5. Tree-of-Thought (Multiple Paths)

For complex decisions, explore multiple approaches:

When a customer seems unsure about coverage amount, consider THREE approaches:

Approach A: Budget-First

- Ask their budget
- Show what coverage fits
- Explain trade-offs

Approach B: Need-First

- Assess their assets
- Calculate ideal coverage
- Show cost to protect

Approach C: Comparison

- Show 3 tiers (basic, standard, premium)
- Explain what each covers
- Let them choose

Evaluate which approach fits based on:

- Their questions (price-focused vs. coverage-focused)
- Their situation (new driver vs. experienced)
- Their communication style (detailed vs. quick)

Then use the best approach.

6. Contextual Adaptation

Adjust style based on customer signals:

Adapt your communication based on customer cues:

If customer uses:

- Technical terms → Match their expertise level
- Casual language → Be friendly and conversational
- Formal language → Be professional and precise
- Short messages → Keep responses concise
- Long messages → Provide detailed explanations

If customer seems:

- Confused → Slow down, use simpler terms, more examples
- Rushed → Get to the point, offer to explain later
- Skeptical → Provide facts, cite sources, be transparent
- Excited → Match enthusiasm, move quickly
- Frustrated → Show empathy first, then solve

Example:

Customer: "idk what i need just give me something cheap"

You: "Got it! Let's find you affordable coverage. Quick question: do you own or lease your car? That'll help me recommend the right basics."

Customer: "I'm evaluating comprehensive vs. collision coverage for my 2023 Tesla Model 3. Can you clarify the deductible implications?"

You: "Absolutely. For a 2023 Tesla, I'd recommend both. Comprehensive covers non-collision events (theft, weather, vandalism), while collision covers accidents. Given Tesla's repair costs, I'd suggest a \$500-\$1000 deductible to balance premium savings with out-of-pocket risk. Would you like me to run scenarios for each?"

✓ Prompt Engineering Checklist

Before sharing with implementers:

- ☐ Personality is clearly defined
- ☐ Required information is listed
- ☐ Conversation flow is specified
- ☐ Edge cases are handled
- ☐ Tone is appropriate
- ☐ Examples are provided
- ☐ Tested with 5+ conversations
- ☐ Refined based on test results

- ☐ Documented any issues
 - ☐ Ready for implementation
-

Key Takeaways

1. **Prompt engineering is iterative** - Your first version won't be perfect
 2. **Be specific** - Vague prompts get vague results
 3. **Test thoroughly** - Try edge cases, not just happy paths
 4. **Provide examples** - Show, don't just tell
 5. **Define boundaries** - Say what NOT to do
 6. **Maintain consistency** - Keep personality and tone aligned
 7. **Collaborate** - Work with implementers to refine
-

Resources

Learn More:

- OpenAI Prompt Engineering Guide: <https://platform.openai.com/docs/guides/prompt-engineering>
- Google AI Studio: <https://aistudio.google.com/>
- Prompt Engineering Course: <https://learnprompting.org/>

Workshop Materials:

- System Prompt Template: [backend/system_prompt.py](#)
 - Test Cases: See workshop guide Part 3
 - AI Studio Guide: [docs/AI_STUDIO_GUIDE.md](#)
-

Workshop Timeline

Part 2: Build Agent Personality (15 min)

0:15-0:20 | Orchestrators Design Prompt

- Open AI Studio
- Use template
- Write initial version

0:20-0:25 | Test & Refine

- Test with 3 conversations
- Note issues
- Refine prompt

0:25-0:30 | Share with Implementers

- Read prompt out loud
 - Implementers code it
 - Test together
-

Remember: You're not just writing instructions - you're designing the agent's personality and intelligence! 🧠