

# Achieving AI Business Value:

From “*Jobs to be Done*” to “*Outcomes to be Eval’d*”

Author: Tom Gersic  
Head - AI and Digital Business

# Executive Summary

In the rapidly evolving landscape of artificial intelligence, businesses often struggle to transform AI investments into tangible value. This paper presents a practical, human-centric approach to harnessing AI effectively, grounded in the Jobs to be Done (JTBD) framework and enhanced by rigorous evaluation-driven development (“Outcomes to be Eval’d”). By clearly defining user needs (the “jobs” that the AI system is “hired” for), emphasizing the human experience, and continuously evaluating AI outcomes through explicit success metrics, organizations can ensure alignment between AI solutions and business goals. Demonstrated through real-world examples and practical demonstrations, this approach ensures that AI projects move beyond technical success to deliver measurable, user-validated business impact.





Introduction:

# Focusing on Humans to Drive AI Value

Artificial intelligence (AI) is everywhere in today's business conversation, often surrounded by hype and lofty promises. Yet the key question remains practical: How do we use AI in a way that delivers real, measurable business impact? In our professional services business, we too frequently see organizations dive into AI initiatives reactively, chasing the latest tools, features, and point solutions without a clear purpose. The result is often wasted effort, a fragmented technology landscape, and disappointing ROI. To avoid this, we ground AI projects in human-centered principles: understanding the people involved, designing

for user experience, and ensuring solutions are well adopted in practice. To do this, we introduce a framework that combines the **Jobs to be Done (JTBD)** (see <https://jobs-to-be-done.com/>) approach with what we might call **"Outcomes to be Eval'd"**, an evaluation-driven twist for AI engineering. The goal is to deeply understand what users need from an AI system to achieve their business objectives, and to start building with those end goals and evaluation criteria in mind from day one.

## Who is this for?

These insights are aimed at business and technology leaders looking to maximize value from AI, as well as our own global team of AI practitioners seeking a common approach. Whether you're integrating OpenAI's latest models into workflows, leveraging Snowflake for data, building AI-driven features in Salesforce, or deploying on AWS, the same human-centered thinking applies.

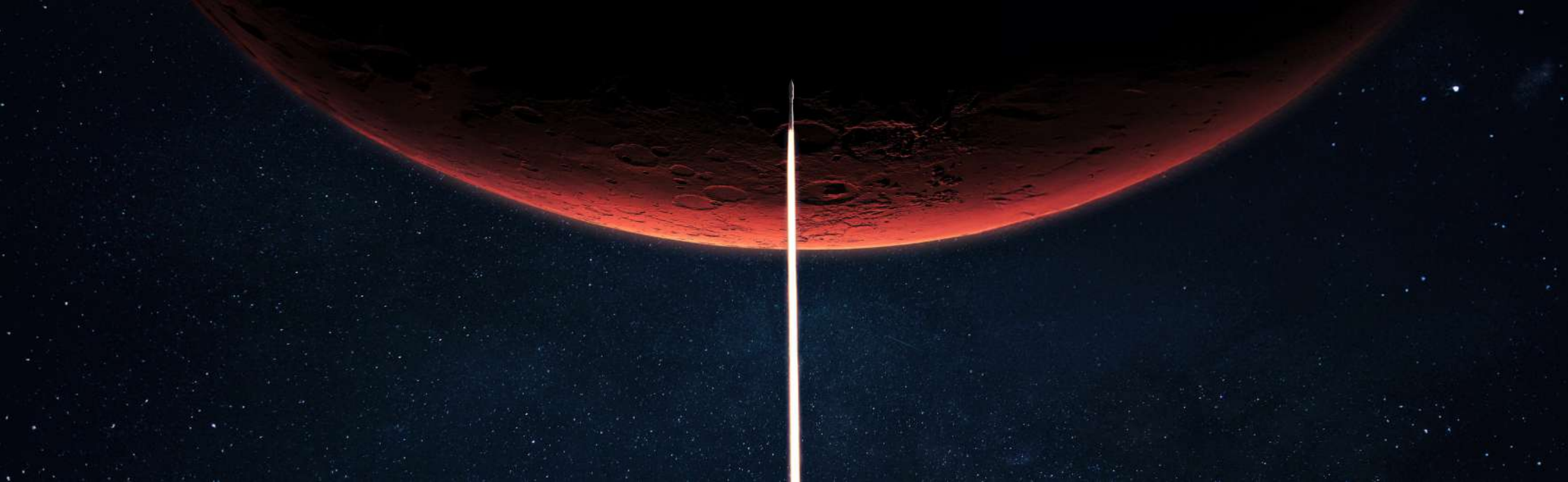
# Jobs to be Done: Understanding the Real “Job” for AI

Jobs to be Done (JTBD) is a framework that shifts the focus from technology or features to the core problem the user or customer is trying to solve. In other words, people “hire” a product or technology to get a specific job done. As Harvard Business School professor Theodore Levitt once said, “People don’t want to buy a quarter-inch drill. They want a quarter-inch hole!” When we apply JTBD to AI initiatives, we first ask: What outcome or task are we expecting AI to accomplish for us or our customers? By articulating this, we ensure the AI solution is anchored to a genuine need rather than a shiny gimmick.

*For example, a business shouldn’t implement AI for its own sake (despite well-publicized C-level AI mandates), it should implement AI to achieve outcomes: “increase operational efficiency to reduce costs” or “enhance customer experience to improve loyalty.” These are the “jobs” the business is hiring AI to do. Framing needs in terms of jobs shifts the conversation from “What can this AI tool do?” to “What result do we need, and how might AI help deliver it?”.*

**Real World Example:** Salesforce’s product teams have long used JTBD to guide innovation, recognizing that customers “*buy or hire products and services to get a specific job done.*” This approach helped them align products to the outcomes customers “hire” those products to achieve. [\(Do You Really Understand What Your Employees Need? This Framework Can Help\)](#)





Similarly, when considering an AI feature, we should ask: *What job is the user hiring this AI to do, and what **outcomes** define success?*

JTBD encourages us to look beyond superficial requests and dig into the why behind them. If someone says, “We need an AI chatbot,” the JTBD approach would probe further: is the real job to provide 24/7 customer support? To reduce call center load by 30%? To improve customer satisfaction by delivering instant answers? By clarifying the desired outcome (the jobs to be done), we discover solutions and targeted AI capabilities to provide these outcomes.

In summary, focusing on the “job” forces AI projects to address real pain points and value drivers.

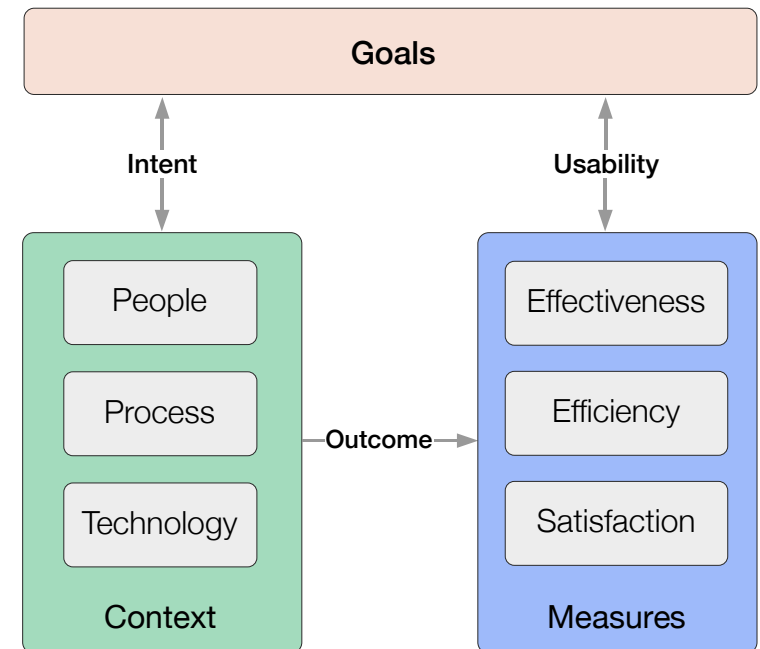
# The User Experience Matters

Identifying the right job to be done is only the beginning. To achieve business value, an AI solution must also be usable and adopted by the people it is meant to help. This is where human-centered design and user experience (UX) become critical. After all, users don't care how advanced the technology is. They care whether it actually helps them in an easy and enjoyable way. A system that technically works but frustrates users will see low adoption, undercutting any promised ROI.

In our experience (and backed by our own client work), successful AI-powered products almost always go through a phase where user experience is the primary focus. Many projects start with impressive tech demos or prototypes, but the breakthrough to wide adoption comes when the design is refined around the user's journey and needs. This involves simplifying complexity, building trust, and addressing fears (for example, ensuring the AI is seen as a helpful assistant rather than a threat to jobs). The key question to measure success becomes: "Is this product adopted by users?" If it isn't, even the most powerful AI will fail to create value.

A human-centered design approach guides us. It means involving users early and often: understand their context and needs, design solutions with their input, and iterate based on feedback. One well-known method is the user-centered design cycle (as defined in ISO 9241), which includes steps to understand users through research, derive requirements from user needs, create prototypes, and then collect user feedback for iterative improvement. By cycling through these steps, we ensure the AI solution remains aligned with human needs at every stage.

For AI systems, human-centered design also involves things like transparency, control, and training: making the AI's actions explainable to users, giving users some control or fallback options (so they feel confident using it), and educating users on how to best interact with the AI. All these UX considerations bolster trust and adoption. In summary, an AI solution optimized for the "job to be done" and wrapped in a great user experience is far more likely to be embraced, delivering its intended business value.



# “Outcomes to be Eval’d”: Designing with the End in Mind

So far, we have the outcome defined (job to be done) and a process to design around users (human-centered design). Now we introduce a complementary concept: **“Outcomes to be Eval’d”** This idea extends JTBD into the realm of engineering and measurement. In essence, **for each “job” we want done by AI, we also define how we will evaluate if that job is done well.** We use those evaluations to drive the development from the start. It’s a mindset of evaluation-driven engineering for AI systems.

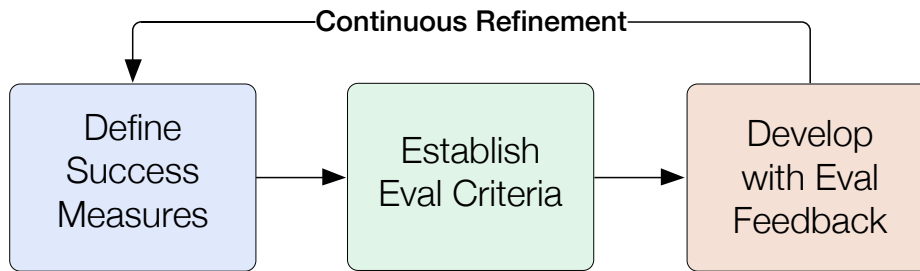
Why is this especially important for AI? Traditional software is usually deterministic: If requirements are met, things just work. AI systems are probabilistic and can behave unpredictably or produce partial errors. This makes testing and validation trickier. If we only evaluate at the end (or not at all), we risk discovering too late that the AI isn’t actually meeting the user’s needs or business goals. Instead, **we evaluate AI output continuously, even as we are developing the system, rather than treating**

**evaluation as an afterthought.** By *starting with the evals* in mind, we clarify what success looks like and catch issues early.

In practice, “Outcomes to be Eval’d” means that once you know the job to be done, you immediately ask: *How will we measure that the AI is doing this job successfully?* For example, if the job is “reduce support ticket resolution time by 30%,” your evaluation criteria might include metrics like average resolution time, customer satisfaction scores, or a comparison against human baseline performance. If the job is “answer customer questions accurately via a chatbot,” the Eval could involve a set of test queries with expected correct answers and measuring accuracy or user ratings of those answers. By defining such evals, you essentially create a **target** for the AI system to hit.

What does this look like in practice? Here’s a simplified roadmap for applying Jobs to be Eval in an AI project





**Define Success Measures:** Based on the job to be done, determine the quantitative and qualitative metrics that indicate success. For instance, if the AI’s job is to triage customer emails, metrics might be accuracy of classification, response time, and user satisfaction rating.

**Establish Eval Criteria:** Determine how the metric should be measured, and set up evaluation tests for the AI.

**Develop Iteratively with Feedback from Evals:** Build the smallest version of the AI system and immediately test it with your evals. Because AI results can vary, you want to catch where it’s failing early.

**Continuous Refinement:** Our work doesn’t stop at deployment. Once in production, continue to monitor those evaluation metrics. AI systems can drift or encounter new edge cases, so a culture of ongoing evaluation and improvement is essential.

In summary, **“Outcomes to be Eval’d” means baking your success criteria and testing right into the engineering process.** It’s the bridge between a great idea (JTBD) and a great result (achieved reliably in practice). By starting with evals, teams are forced to be explicit about what “good” looks like, which greatly increases the chance of delivering a system that actually works as intended. It also makes debugging and improving the AI more systematic. You know when you’re moving in the right direction because your eval scores improve.



## Implementation Scenario: An Employee Asks for Company-Recognized Holidays

Employees frequently will want quick access to company-recognized holidays to effectively plan personal events, coordinate vacation days, and avoid conflicts with important business meetings or deadlines. In this scenario, an employee interacts with an AI-powered HR assistant to ask for the official list of company holidays. The AI assistant must promptly deliver accurate, personalized holiday information derived directly from the official HR Employee Handbook.

Using the Jobs to be Done (JTBD) framework, we define the task clearly with a Core Job, Job Steps, and Expected Outcomes that we will use to create Evals for our AI Agent:

Core Job:

“

*When planning my schedule, I need to quickly check the company's official holidays, so I can coordinate my personal plans without conflicts.*

”

## Job Steps:

1. Define the question clearly.
2. Access the authoritative HR policy source.
3. Provide a simple, accurate list of holidays.

## Outcomes to be Evaluated:

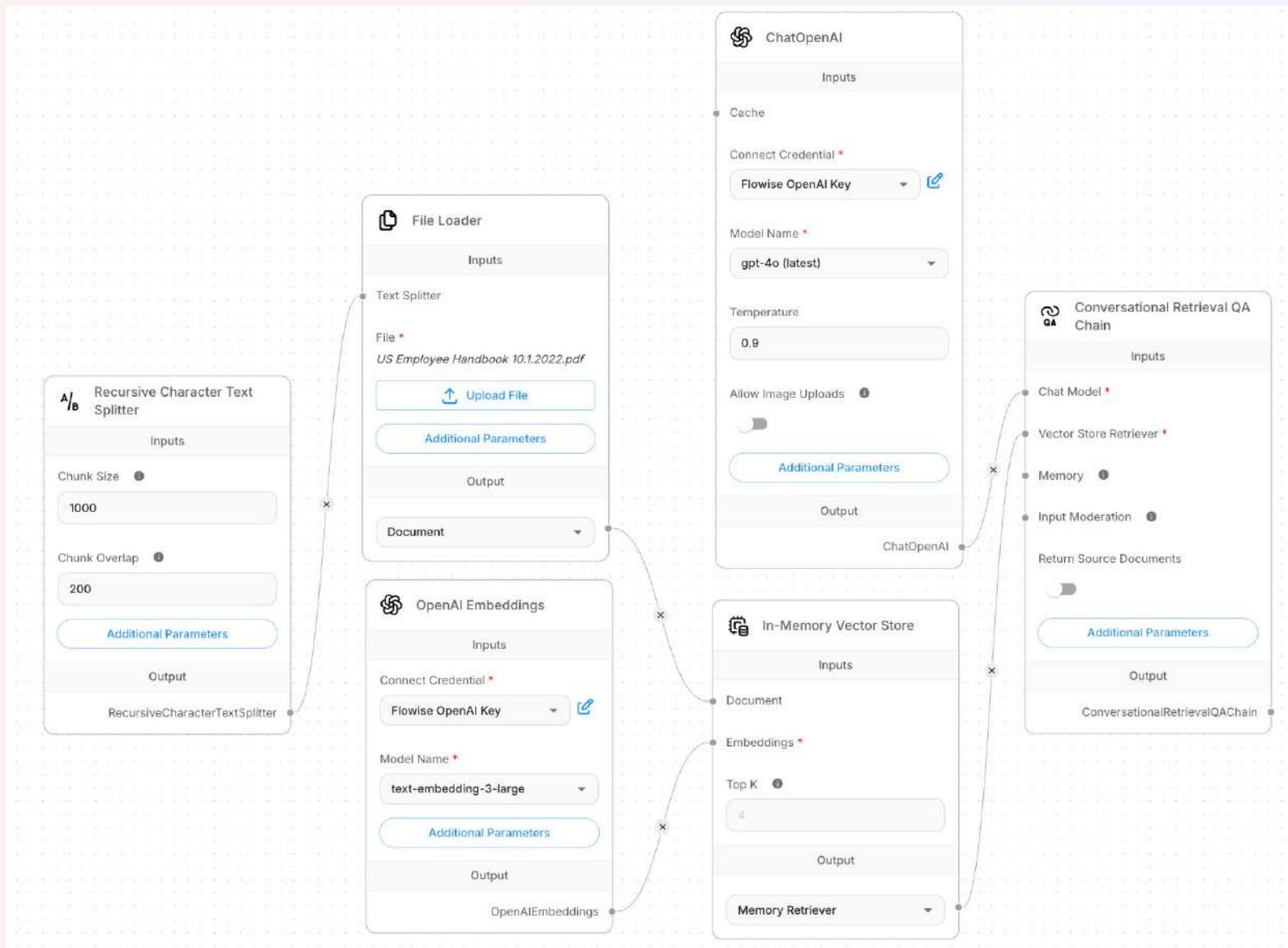
**Accuracy:** The response matches key holiday dates in the HR Employee Handbook (e.g., “New Year’s Day,” “Thanksgiving,” “Christmas”).

**Latency:** The response time measured from user query to AI-generated answer is less than 1 second.


## Demonstration:

To demonstrate this concept, we will implement a very simple RAG system using Flowise, an open-source, low-code platform that enables users to visually build, prototype, and deploy workflows powered by large language models (LLMs) via a drag-and-drop interface. Flowise is selected for its ease of use for this scenario. Other more well-rounded Eval frameworks include OpenAI Evals, Langsmith, and Langfuse.


As you can see below, an example HR Employee Handbook is ingested into an In-Memory Vector Store in Flowise using OpenAI’s text-embedding-3-large model.




Once the document is chunked, embedded, and loaded into the vector store, a user can manually ask questions with the Conversational Retrieval QA Chain node, using OpenAI's GPT 4o model:



Hi there! How can I help?

What holidays are recognized?

The recognized holidays are:

- New Year's Day
- Martin Luther King Jr. Day
- Memorial Day
- Independence Day
- Labor Day
- Thanksgiving Day
- Day after Thanksgiving
- Christmas Day

*This manual interaction is exactly how an end-user will interact with this system, but how can we ensure that this answer meets the expected outcomes in a more systematic and “Eval Driven” way? To show this, we will create a sample test dataset that gives the expected result to a user’s question regarding recognized holidays:*



←

Dataset : HR Employee Handbook Q&A

⬆️

Upload CSV

+

New Item

Input

Expected Output

What holidays are recognized?

New Year's Day Martin Luther King Jr. Day Memorial Day Independence Day Labor Day Thanksgiving Day Day after Thanksgiving Christmas Day

↕

⋮

We will then create evaluators that will engage with the chat agent, and confirm that the expected outcomes are met:



#### Accuracy:

**a. Keyword Pass/Fail:** Check AI response against HR Employee Handbook key holiday dates (e.g., “New Year’s Day,” “Thanksgiving,” “Christmas”).

**b. LLM Judge Model:** GPT-based judge scores accuracy on a scale (0-1), validating completeness and correctness.









#### 2. Performance:

**a. Latency:** Response time measured from user query to AI-generated answer. Target threshold: < 1 second.

## Evaluators

Search [ ⌘ + F ]

+ New Evaluator

| Type   | Name   | Details   | Last Updated  |
|--|--|---|---|
|  Numeric    | HR Holiday Chatflow Latency less than 1 second | <b>Measure:</b> Chatflow Latency<br><b>Operator:</b> Less Than<br><b>Value:</b> 1000  | August 3rd 2025, 03:15 PM  |
|  LLM Based  | Holiday Grader                                 | <b>Prompt:</b> Evaluate the degree of hallucination in the generation on a continuous scale from 0 to 1. A gener...<br><b>Output Schema Elements:</b> score, reasoning                        | August 3rd 2025, 03:10 PM  |
|  Text Based | HR Holiday Evaluator                           | <b>Operator:</b> Contains All<br><b>Value:</b> New Year's Day, Martin Luther King Jr. Day, Memorial Day, Independence Day, Labor Day, Thanksgiving Day, Day after Thanksgiving, Christmas Day | August 3rd 2025, 03:09 PM  |

We can then run the Evaluation to confirm that the agent is performing as expected. In this case, you'll see that the answer was accurate (pass), but the latency was too high (fail).

## Evaluations



+ New Evaluation

| <input type="checkbox"/> | Name                        | Latest Version | Average Metrics   | Last Evaluated           | Flow(s)                       | Dataset                  |  |
|--------------------------|-----------------------------|----------------|---|--------------------------|-------------------------------|--------------------------|--|
| <input type="checkbox"/> | Recognized Holidays Latency | 1              | <div>Total Runs: 1</div> <div>Avg Latency: 5045.880ms</div> <div>Pass Rate: 0.00%</div>   | 03-Aug-2025, 03:17:42 PM | Memory Vector Store Eval Test | HR Employee Handbook Q&A |  |
| <input type="checkbox"/> | Recognized Holidays         | 1              | <div>Total Runs: 1</div> <div>Avg Latency: 4010.780ms</div> <div>Pass Rate: 100.00%</div> | 03-Aug-2025, 03:10:28 PM | Memory Vector Store Eval Test | HR Employee Handbook Q&A |  |

The LLM Eval scores the answer with an “LLM as a Judge” method, giving a score (0 is best, 1 is worst) and reasoning for the score. Using this method we can ensure that the system is giving a good answer using a repeatable and systematic process for evaluation.

| Expected Output   | Memory Vector Store Eval Test   |                      |   |
|---|---|----------------------|---|
|   | Actual Output   | Evaluator            | LLM Evaluation  |
| New Year's Day Martin Luther King Jr. Day Memorial Day Independence Day Labor Day Thanksgiving Day Day after Thanksgiving Christmas Day | <p>The recognized holidays are: - New Year's Day - Martin Luther King Jr. Day - Memorial Day - Independence Day - Labor Day - Thanksgiving Day - Day after Thanksgiving - Christmas Day</p> <div> <div>Total Cost: \$ &lt;0.01</div> <div>Total Tokens: 1025</div> <div>API Latency: 4010.78</div> </div> | HR Holiday Evaluator | <div>SCORE: 0.3</div> <div>REASONING: The generation contains some minor discrepancies that contrast with established information but mostly aligns with factual data and reasonable inference.</div> |

# From Hype to Human-Centric Success

Adopting AI in business is not about jumping on the latest trend; it's about solving real problems and **augmenting human capabilities** in meaningful ways. By starting with the **Jobs to be Done** perspective, we ensure every AI project is grounded in a clear purpose. By adding a **“Outcomes to be Eval'd”** approach, we commit to defining what success looks like and tracking it systematically, which keeps the project on course toward that purpose. And by following human-centered design principles throughout, we make sure the solution actually works for the people who use it, encouraging adoption and trust.

This framework is especially valuable for implementation partners like us (whether working with OpenAI's models, Snowflake data platforms, Salesforce applications, or AWS infrastructure) because it provides a common language between business stakeholders and technical teams. It bridges the gap from *idea* to *impact*. We start with empathy for the user's job, and we end with proven value, with evaluation scores and user feedback as our guideposts along the way.