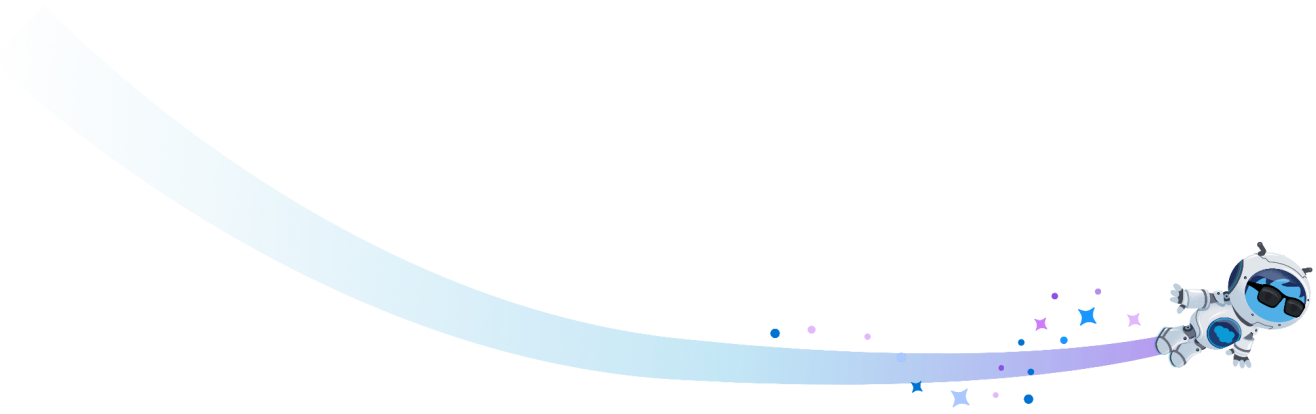
Agentforce Implementation Guide

Employee-Facing Agents



Last Updated: November 3rd, 2025

Table of Contents

Get Started with This Guide

In this guide, you'll learn how to launch a employee-facing agent using the Agentforce Employee Agent template. This guide walks you through the process of creating an agent by following the five major stages of agent development: Ideate, Build, Test, Deploy, and Monitor. You can use this guide for a variety of solutions:

Goal	Instructions
I just want to build and launch an agent fast.	Skip the Ideate chapter and use the Build, Test, and Deploy chapters to launch an agent. After launch, you can review the Monitor chapter to refine your agent.
I'm new to building agents.	Walk through the entire guide in order.
I need ways to monitor and improve my agents	Review the Test and Monitor chapters in order to refine your agent.

What are Agents?

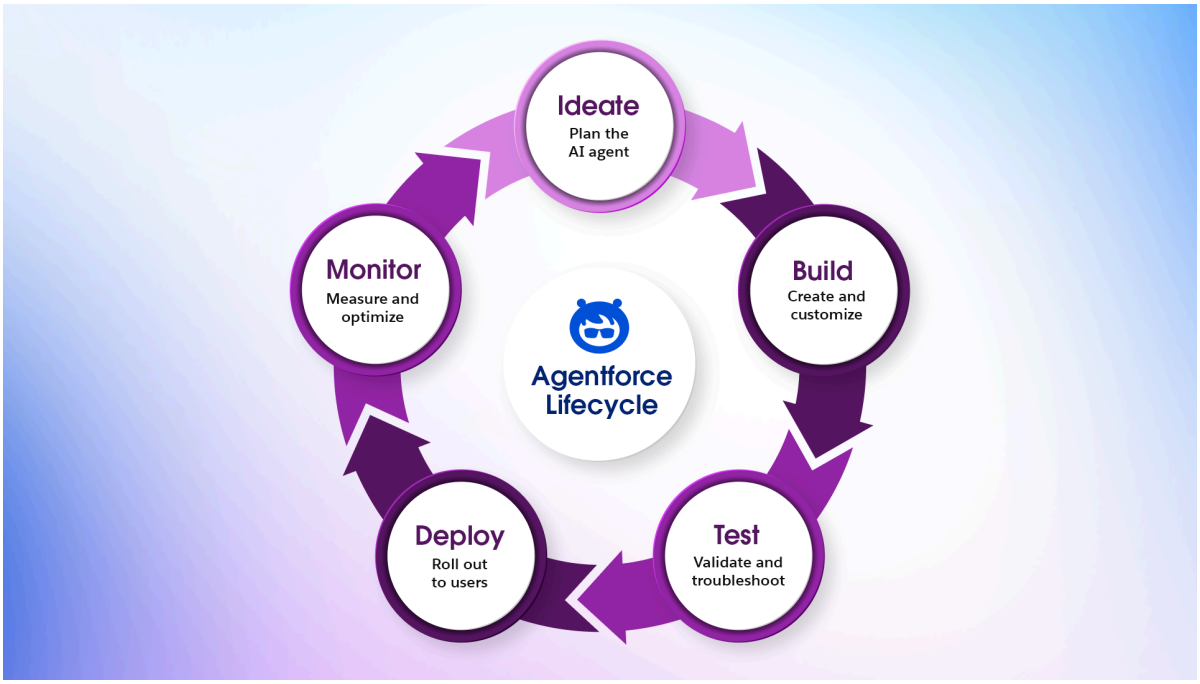
Agents are goal-oriented, autonomous AI assistants that perform tasks and business interactions. They can initiate and complete a sequence of tasks, handle natural language conversations, and securely provide relevant answers drawn from business data.

Some agent types are best at assisting and collaborating with a Salesforce user in the flow of work. Other types can also act on behalf of a user or customer, based on the use cases and guardrails an admin specifies. Agents adapt and improve with each interaction. Some agent types are available in your Salesforce interface, and you can add some to your customer channels.

Agents respect standard Salesforce access controls such as licenses and permissions, so they always act securely. Agents also include AI guardrails designed to help meet ethical standards and are integrated with the Einstein Trust Layer, a secure AI architecture natively built into the Salesforce Platform.

The Agent Development Lifecycle

The development lifecycle for an AI agent generally follows five key stages: plan, build, test, deploy, and monitor. Here's an overview of what happens during each stage.



- **Ideate**: Define the agent's business value use case, scope, and required capabilities. Address concerns and risks upfront by integrating risk management into the planning process to ensure the agent meets ethical, legal, regulatory, and security requirements.
- **Build**: Create and customize the agent, integrating necessary data sources, and training its underlying model(s) on the Agentforce platform.
- **Test**: Validates the agent's performance, accuracy, and adherence to security and compliance standards. Troubleshoot and resolve any issues.
- **Deploy**: Configure the agent's connections and make the agent available to users on customer support channels.
- **Monitor**: Measure the agent's performance and health, collecting feedback and data to identify areas for optimization and continuous improvement.

This guide walks you through each stage of the agent development lifecycle.

A Real World Example

In this guide we follow a fictional company, Makana Medical Devices, as they work step-by-step to implement an Agentforce Employee Agent in their organization.

Makana Medical Devices is a medical device company that develops, manufactures, and distributes a wide range of diagnostic, treatment, and patient monitoring instruments. They've already integrated generative AI into their Salesforce ecosystem and are now looking to build their first employee agent with Agentforce to help their sellers manage opportunities in Salesforce.

Sign Up for a Salesforce Org

To follow along with the steps in this guide, you can sign up for a free Salesforce Developer Edition org.

1. Go to the Developer Edition signup page.
2. Fill out the form and submit it.
3. Look for a welcome email from Salesforce in your inbox.
4. Verify your account using verification link in the email.
5. When prompted, set a new password and select a security question for your Developer Edition org. Then complete the login process.
6. Make sure to keep track of your username and password so that you can log in again later from the Salesforce login page.

When you're done signing up for a Salesforce org, you can start the agent development process.

Ideate Your Agentforce Solution

The first step in developing an AI agent is ideating your Agentforce solution. Planning an AI agent is crucial for several reasons. It ensures that your AI project aligns with your company's overall AI strategy and business goals. It also helps you define the agent's work and assess its potential impact on the organization.

Addressing concerns and risks upfront by integrating risk management into the planning stage can accelerate projects and ensure they meet ethical, legal, regulatory, and security requirements. Without proper planning, an agentic AI project might fail to deploy or be scrapped entirely.

When planning an agent, here are the things to consider:

- Use case definition and scope
- Data and technical requirements
- Risks and guardrails
- Business processes and agent design

Although careful planning is important, try to strike a balance between experimentation and planning. When you're getting started with Agentforce, hands-on learning and prototyping are essential for understanding AI agents and testing out ideas.

In this chapter, we walk through an ideation exercise for Makana Medical Devices. If you want to get straight into creating an agent, skip ahead to the Build chapter.

Identify Your Use Case

An Agentforce use case is an application of AI technology where an AI agent takes an action or series of actions that accomplish a goal or job-to-be-done on behalf of your employees, customers, or organization.

As you begin collecting ideas for Agentforce use cases–whether through crowdsourcing, business process review, user research, or market analysis–you might notice that some ideas lend themselves well to autonomous AI applications, and some don't. When evaluating potential use cases for autonomous AI, consider the following questions:

- **Value**: Why are you delegating this work to an AI agent? Is the agent faster or more accurate? Will it provide a better experience?
- **Work**: Can you describe the work the AI agent will do? Do you fully understand all the business processes involved in that work?
- **Decision-making**: Does the work involve decisions and steps that can be completed without direct human input or judgment? Are there well-defined policies, rules, and constraints that the AI agent can follow alone?
- **Risk**: Can the AI agent operate within the security, legal, ethical, and regulatory requirements that apply to the work?
- **Data**: Is the data capable of supporting the work the AI agent will do?

As you decide on your answers, it's important to understand what agents can do. Often the easiest way to get started is by building an agent and seeing how its capabilities match with the goals you have for implementation.

Makana's Business Challenge

Makana Medical Devices' sales reps often spend a significant amount of time on manual data entry, navigating workflows, and performing tasks related to creating and updating opportunities in Salesforce. This takes away valuable time that could be spent on core selling activities, such as engaging with prospects, building relationships, and closing deals.

By implementing an AI agent to assist with opportunity management, Makana aims to:

- **Reduce administrative overhead**: Automate opportunity creation, data population, and task assignment. This frees up sales reps to focus on high-value activities and ensures data accuracy.
- **Accelerate the sales cycle**: Streamline the process of creating and managing opportunities, leading to faster progression through the sales pipeline.
- **Enhance sales rep productivity and effectiveness**: Empower sales reps with tools that make their workflow more efficient and allow them to dedicate more time to selling.

Makana wants to help sales reps be more productive. They also want to optimize the opportunity management process in Salesforce. Both of these will improve sales outcomes. They come up with a few use case ideas for their Agentforce Employee Agent, and the two most promising ideas are: automating opportunity creation and assigning next steps.

Define Your Agentforce Use Case

After identifying your use case ideas for Agentforce, flesh out each of the use cases so your organization can assess and prioritize them. Remember, this stage of the planning process is about the project goals, not the technical solution.

Identify the Jobs To Be Done

First, describe the work the AI agent will do. Many organizations use the Jobs to Be Done framework to outline the role of the agent and the tasks it will perform. Be sure to think deeply

about the work and its expected outcomes. It's an essential step in understanding how the agent can impact your organization, customers, and employees.

Here are the jobs to be done for Makana's top use case idea.

Work or Job to be Done	Tasks
Opportunity Management	• Create opportunities and identify deal resources • Create tasks to connect with deal resources • Summarize previous calls and identify next steps • Create a meeting brief • Identify cross-sell and up-sells • Generate best next sales actions • Create and update quotes • Suggest pricing strategies based on customer data and historical deal performance

Determine the Scope

After identifying the work you want the AI agent to do, the next step is figuring out the right amount of work. What's the appropriate scope? Your minimum viable product (MVP) should be the smallest unit of work that delivers value to your employees. An iterative approach allows you to validate your assumptions, demonstrate value, manage the level of risk, and develop a plan for scaling the AI solution. Gradually expand your scope and use cases as your organization evolves in AI maturity.

Let's take a look at how Makana might scope its use cases for Opportunity Management.

Work or Job to be Done	Tasks
Opportunity Management	Version 1 (MVP): • Create opportunities and identify deal resources. • Create tasks to connect with deal resources. Capabilities Added to Version 2: • Summarize previous calls and identify next steps. • Create a meeting brief. Capabilities Added to Version 3: • Identify cross-sell and up-sells. • Generate best next sales actions. • Create and update quotes. Capabilities Added to Version 4:

	• Suggest pricing strategies based on customer data and historical deal performance.

Define the Business Value

When you fully understand the scope of the work the AI agent will perform, you can establish the business value of the use case. Be sure to set specific, measurable goals and focus on outcomes.

Here's the business value for Makana's use cases, which aligns with the company's top objectives.

Work or Job to be Done	Tasks
Opportunity Management	• Save 5 minutes effort on each opportunity for sales reps • Increase sales rep satisfaction by 10%

Evaluate Data Readiness

Next, evaluate your data readiness for the use case. For AI agents to perform optimally, they must be powered by trustworthy, high-quality data that's relevant to the business context. Don't commit to a use case until you've confirmed that the data can support it.

Makana's AI implementation team does a cursory investigation into their data readiness for the FAQs and Case Management use cases, and here's what they find.

Use Case	Data Sources	Data Readiness
Opportunity Management	Salesforce CRM	Makana uses Sales Cloud, and they have good data governance practices in place. The data quality in Sales Cloud is sufficient for the opportunity management use case.

Keep in mind that when you're defining your use case, you're not conducting a full-scale data audit. It's more of a preliminary gut-check to determine the use case's feasibility and ease of implementation. You'll dig deeper into data readiness when you define the technical requirements for the project.

Assess and Prioritize the Use Cases

After defining your Agentforce use cases, your organization can assess the feasibility and impact of the projects and then incorporate them into your prioritized AI roadmap. See AI Strategy for more information about the factors to consider when refining your AI backlog and which prioritization frameworks you can use.

Define the Technical Requirements

When planning an AI agent, spend some time gathering technical requirements up front so you can build the right solution faster. Key factors to explore include: data, channels, routing and escalation, and security controls.

Data

Your organization needs to be data-ready to start an AI project, which means the data for the project is accurate, complete, available, accessible, and securely governed. When defining the data requirements for your Agentforce use case, be sure to assess the data quality, migrate and integrate data, establish data governance, and create a plan for data analytics. To learn more about data requirements, see AI + Data: Project Planning.

Channels

Consider the channels where the AI agent will operate. How will the agent engage with employees or customers? How will the agent be presented to users? See Deploy Your Agent to Channels to learn more about the different options for deploying agents to channels.

Makana's new employee agent will be deployed on the following channels: the Salesforce desktop app, the Salesforce Mobile app, and Slack.

Keep in mind that you can prototype an AI agent in your sandbox before configuring the channels if you want to test out your ideas for the agent's design. But you need to thoroughly test the agent's performance on each channel before deploying to production.

Routing and Escalation

AI agents sometimes need to escalate conversations to a live representative. Some reasons for escalating to a rep can include company policy, brand requirements, security measures, risk management, or regulatory compliance.

During the planning process, define all the ways in which human decision-making and oversight will occur during the AI agent's work. Clear guidelines about when and how reps should step in will make it easier for your organization to configure the right guardrails as you start prototyping your agent.

Security Controls

When you start planning your AI agent, consider permissions and access for admins, employees, customers, and the AI agent itself.

- **Admin Access**: To create and manage AI agents in Salesforce, you need the **Manage AI Agents** user permission and the required permissions for your agent type, or the **Customize Application** user permission.
- **Employee Access**: If your employees interact with the AI agent in Salesforce, the agent runs in the context of the currently logged-in Salesforce user. See Trust and Agents.

Salesforce Considerations for Agentforce Projects

Your AI agent must coexist with your current Salesforce configuration and architecture. Consider Salesforce-specific factors that can influence your Agentforce implementation.

Licensing and Provisioning

Each Agentforce project is unique, so the Salesforce products and features you need vary depending on your use case. Check with your Salesforce account executive to confirm the licensing requirements for the specific AI agent you want to build.

Billing and Consumption

Agentforce uses a consumption-based pricing model. For more information on how usage is billed, see Agentforce Pricing. You can also refer to your contract or contact your account executive.

Requirements

To use Agentforce, these products and features must be enabled in your Salesforce org.

- Lightning Experience
- Einstein Generative AI
- Data Cloud

Existing Automation

With Agentforce, your AI agents use agent actions as tools to accomplish specific tasks. Those agent actions are built on top of existing Salesforce Platform technology, such as flows, Apex, and prompt templates. Start cataloging any existing automations related to your use case that you might be able to repurpose for your AI agent.

Define the Agent's Guardrails

Autonomous AI has inherent risks that you can mitigate including security threats, data breaches, reputational harm, financial loss, bias, hallucinations, and issues with transparency and accountability. The good news is by building agents on the Agentforce platform, you have access to built-in guardrails and controls to help mitigate risk. When planning an AI agent, it's important to discuss potential risks with key stakeholders for your Agentforce use case and use the platform to plan risk mitigation strategies. By creating a robust guardrail strategy, your team can be confident that the solution is addressing the most common AI risks.

Frame Conversations About Risk

As you approach conversations in your organization about AI risk, you can use the People, Business, Technology, and Data framework. These categories and considerations can help you come up with possible risks and concerns related to your Agentforce project.

Category	Considerations

People	• **Empowerment**: Roles and responsibilities, hiring, training, and upskilling • **Culture and practice:** Human-centered AI design, change management, adoption
Business	• **Value**: Benefits, objectives, KPI, and metrics • **Operations**: Org structure, capability management, processes and workflows, AI governance, DevOps strategy
Technology	• **AI tooling**: AI infrastructure, applications, APIs, prompts, security safeguards • **AI models**: Model selection, training considerations, management, cost
Data	• **Quality**: Fit for use, accuracy, completeness, accessibility, recency, and more • **Strategy**: Data management, infrastructure, governance, analytics

Identify Risks and Concerns

Makana uses this framework to discuss risks related to their use cases. Let's take a look at the risks and concerns that Makana's stakeholders identified for each category.

Category	Risks
People	• **Rejection**: Customers don't want to talk to the agent because they don't trust it or because they're unsure if they're allowed to use AI. • **Abuse**: Customers are hostile to the agent or try to manipulate it. • **Culture**: Fears about the potential impact of AI on service jobs affect employee morale.
Business	• **Fit**: Agent's scope doesn't fit properly into the business organization or team processes. • **Reporting**: Current team KPIs are invalidated by the introduction of an AI agent to do some of the work. • **Incentives**: Compensation and reward structures are impacted by agent work redirection. • **Operations**: Process for escalation is unclear, inefficient, or frustrating. • **Agent Performance**: Appropriate company policies don't correctly influence the AI agent responses. • **Liability**: Financial and reputational damage due to lawsuits or regulatory fines resulting from potential AI errors.
Technology	• **Accuracy**: Hallucinations degrade the quality of responses or endanger the wellbeing of patients.

	• **Reliability**: Variability of the agent's generated responses is too broad. • **Audit**: Technology operations can't track the accuracy of agent responses. • **Latency**: Agent can't achieve timely responses.
Data	• **Access**: Data permissions aren't understood or enforced; data might be exposed to customers. • **Privacy**: The required data can't be used according to the privacy policy. • **Compliance**: It's unclear if any customer contractual constraints apply to the data; for example, maybe data can't leave the customer's business country. • **Fit for purpose**: Data isn't aligned with the agent's objective, or data rights aren't aligned to the use case. • **Ethics**: Bias in model data could generate inappropriate responses.

Note that this list isn't exhaustive, and every use case involves its own unique risks and concerns.

Define Risk Mitigation Strategies

After cataloging the risks and concerns, come up with mitigation strategies for each risk. As you brainstorm potential guardrails, categorize each guardrail to designate whether it's related to people, business, technology, or data.

Here's an example of potential guardrails for one of the risks that Makana identified.

Category	Risk	Potential Guardrails
People	**User Rejection**: Users don't want to talk to the agent because they don't trust it.	• **People guardrail**: Create a communication strategy and conduct education briefings for customers. • **Technology guardrail**: Design the agent to be transparent about the fact that it's AI. • **Technology guardrail**: Configure a welcome message for the agent that sets the right expectations about its capabilities and how it can assist.

When you're done with the risk mitigation exercise, document the risks and guardrails for your use case. Capturing risk mitigation activities is important for regulatory compliance, useful for internal audits, and will build trust with key stakeholders.

Design Your Agent

The Makana team is making great progress with their AI agent. They defined the Agentforce use case, considered the project's requirements, and developed some risk mitigation strategies. Now it's time to start designing the agent and thinking about how it will be configured.

Map Business Processes

When planning an AI agent, it's important to define the work the agent will do. One way to define the work is to map the business processes related to your Agentforce use case.

Process mapping is a visual representation of the steps in a business process from start to finish, typically using diagrams like flow charts or swimlanes. It helps define the work an AI agent will do by depicting the sequence of steps, start and end points, and required inputs and data.

As you outline the business processes you want to delegate to an agent, consider these questions.

- What goal is the user trying to accomplish?
- How might the user express their goal or intent? How would they phrase their questions or requests?
- What processes are involved in achieving the goal?
- When does each process begin and end? What steps need to be taken? In what order?
- What company policies, rules, and guidelines apply to the process? What guardrails are relevant to each step?
- At what points in the process are decisions made and how?
- What clarifying or follow-up questions might the agent need to ask?
- When would a user need to confirm something before a step proceeds?
- What data is required? What's optional?
- What input do you need to collect from users? Does that input need to be in a specific format or syntax?
- What are the outcomes of each step or process? How is the output used? How is success measured?
- Are there situations when specific language should be used when communicating with users?
- What questions or paths of conversation should never be pursued?
- Under what conditions should the conversation be escalated to a live service rep?

Even if you don't create formal diagrams for each of your business processes, it's still important to think deeply about the work the agent will do so that you design the right solution.

Identify the Topics and Actions

After determining your agent's use case and the work the agent will do, you can identify the right topics and actions for the agent. Agents are made up of topics, which define the different jobs an

agent can do. Topics also contain a set of actions, which are the tools the agent can use to do its job.

Salesforce provides some standard topics and actions for Agentforce out of the box, so you can get up and running quickly. But you can also create custom topics and actions to give your AI agent additional abilities, so it can perform tasks specific to your business.

When identifying the topics and actions for your agent, there are two possible approaches: top-down and bottom-up.
In the **top-down approach**, first identify the topics, then define the relevant actions for those topics.
In the **bottom-up approach**, first list all of the individual actions for your agent, then group them into related topics.

If you've identified the jobs to be done and the relevant tasks, the top-down approach will work best. For an example of the bottom-up approach, see Identify the Topics and Actions in Salesforce Help.

Makana's Topics and Actions
For version one of Makana's employee agent, the agent will help sellers create opportunities and relate them to the relevant account. The agent will also identify other roles in the company who should be involved in the opportunity and automatically create related tasks for the seller.
The agent will accomplish this through a single custom topic, Opportunity Creation, that contains three actions:

- A custom flow-based action called Create Opportunity that collects information from the seller, creates the opportunity record, and links it to the account.
- A custom prompt template-based action called Identify Recommended Sales Team that identifies relevant specialists in other roles within the company that can help with the sale and provides a reason for why they should be involved. For example, a business value manager or solution engineer.
- The standard Create a To-Do action, which creates tasks for follow-up or assignments to complete.

Design the Topics and Actions

After identifying the right topics and actions for your AI agent, start planning how to build them. Following best practices when designing the topics and actions can help your agent perform reliably and effectively.

As you plan your topics and actions, simultaneously build and test your AI agent in a sandbox environment. Try different approaches to figure out what works and what doesn't. The key is to use prototyping and continuous refinement to guide your planning and development efforts.

Design the Topics

Topics are an important component of Agentforce because they define the AI agent's goals and provide the context and direction it needs to achieve those goals.

Learn More: *Check out this video to learn how to create a custom topic. For more information about designing topics, review the different parts of a topic and follow the best practices for topic instructions.*

Design the Actions

If none of the standard agent actions suit your needs, you can build custom actions for your AI agent.

When you create a custom action, you build it on top of existing platform functionality that you want to make available in Agentforce–invocable and REST Apex classes, autolaunched flows, prompt templates, external services, and MuleSoft APIs. In Agentforce, that underlying functionality is called a reference action.

Design Considerations for Reference Actions

Here are some factors to consider when approaching the design of the underlying reference actions for your agent actions.

Deterministic or Prompt-Based

When developing the underlying platform functionality for your agent actions, first review the business processes and tasks related to your use case. Then decide whether the process or task should be deterministic, prompt-based, or a combination of both approaches.

- **Deterministic**: Uses an invocable or REST Apex class or autolaunched flow to generate output. Actions based on flows or Apex are deterministic and use business logic and rules to produce a consistent outcome.
- **Prompt-based**: Uses one or more prompt templates to generate output. A prompt-based action lets you control how a response is written or use reasoning and generative capabilities of an LLM. For example, to generate a summary or perform sentiment analysis, you need to use a prompt template as a reference action. Prompt templates are also used to ground an agent in data, such as knowledge or external system data.

Keep in mind that an action can combine both deterministic and prompt-based approaches. For example, an Appointment Management topic might include an action that cancels appointments. When a user asks to cancel their appointment, the flow-based action is triggered to complete the cancellation. At some point during that flow, the agent could also launch a prompt template that summarizes the user's reason for cancelling.

Inputs and Outputs

In Agentforce, each agent action must have at least one input, which means the underlying flow, Apex, or prompt template must also have at least one input. For example, to look up a hotel

reservation, the input might be the guest's email or reservation number. During a conversation, the AI agent has the autonomy to gather information and decide if it has all the details required for it to trigger the action and pass in the input.

Each action must also have at least one output. The way you build the action determines what the output is, how it's used, and whether and how it's displayed to users in the conversation. Don't be afraid to experiment with test actions that aren't fully implemented–they can be a great way to see how your ideas work in execution.

A Strong Foundation

Now you know that planning and designing an AI agent is all about laying down a strong foundation with creative ideation and thorough planning. You need to identify your use case and objectives, think about your data strategy, consider the user experience, and outline your project's technical requirements. Address potential risks and define your business processes to make sure the AI agent aligns with your organization's operational, security, legal, ethical, regulatory requirements.

Don't make the mistake of taking a "waterfall approach" to agent design. You don't have to develop and deploy with a linear, phased plan. As you consider your Agentforce solution from all the necessary angles, get hands-on, and begin prototyping the AI agent in your sandbox environment. That way, you don't invest too much time up front on a plan that might not work out in the end.

Now that you've planned out and experimented with a new AI agent, let's start making it a reality and learn how you can build an agent that demonstrates real value to your business.

Build Your Agent

Now that Makana has completed their ideation stage, it's time to build the agent. In this stage, you create an Agentforce Employee agent and customize it to better assist Makana's sales reps.

Here's what you'll cover.

- Prepare your org for Agentforce by verifying Data Cloud, setting up Einstein generative AI, and enabling Agentforce.
- Create an agent from the default template for the Agentforce Employee agent type and customize key settings.

- Configure the agent with a custom topic, including a custom flow-based action, a custom prompt template-based action, and a standard agent action.
- Give users access to the agent.

Review Required Editions and Permissions

Before you get started, review the editions and permissions required to complete the tasks in this guide. The Admin Permissions Needed list covers all of the permissions, permission sets, and permission set licenses you'll need to build, test, deploy, and monitor your agent.

REQUIRED EDITIONS
Available in: Lightning Experience
Agentforce Employee Agent is available in: Enterprise, Performance, Unlimited, and Developer Editions with Foundations or Agentforce 1 Editions. Access to some standard agent actions requires additional add-on licenses. Requires Flex Credits.
Slack is available in: All Slack plans with a Salesforce Agentforce license

ADMIN PERMISSIONS NEEDED	
User Permissions	• Customize Application • Manage AI Agents • Manage Flow
Permission Sets	• Data Cloud Architect • Prompt Template Manager Required for Analytics & Optimization (Not Yet Available) • Access Agentforce Optimization • Tableau Next Limited Consumer
Permission Set Licenses	• Data Cloud • Einstein Prompt Templates
Slack Roles	• Workspace Owners/Admins (Free, Pro, and Business+ plans) • Org Owners/Admins (Enterprise plans)

Set Up Einstein Generative AI

Agentforce is integrated with Data Cloud and generative AI features so you can build observable and trusted agentic experiences. We'll verify and set up these features to prepare your org for your first agent.

Verify Data Cloud in Your Org

Data Cloud is required for essential Einstein generative AI and Agentforce functionality, such as the Trust Layer, agent event logs, and consumption billing tracking.

If you're preparing a free Agentforce Developer org, Data Cloud is provisioned and enabled for you. We recommend verifying your Data Cloud installation. From Setup, in the Quick Find box, enter `Data Cloud`, and then select **Data Cloud Setup Home**. Verify that a home org has been created for you on the Data Cloud Setup Home page, with a home org ID, a home org instance, and a tenant endpoint.

If you aren't using a free Agentforce Developer org, or if Data Cloud installation wasn't successful, turn on Data Cloud.

1. From the Setup Menu, select **Data Cloud Setup**.
   If you don't see this option, refresh your page, or log out and then log back in with your admin user credentials.
2. To enable Data Cloud, click **Get Started**.

Setup can take 15-60 minutes. After setup is finished, you can continue on to the next section.

Enable Einstein Generative AI

Turn on Einstein to use generative AI features, including Agentforce, across Salesforce.

1. From Setup, in the Quick Find box, enter `Einstein Setup`, and then select **Einstein Setup**.
2. Enable **Turn on Einstein**.

After you turn on Einstein, it can take a few minutes to sync Einstein and Data Cloud.

Verify Einstein Trust Layer Settings

Build trust in your agents and other generative AI features. Use the Einstein Trust Layer to personalize data privacy controls that are integrated into the end-user experience.

From Setup, in the Quick Find box, enter `Einstein Trust`, and then select **Einstein Trust Layer**.

Einstein Trust Layer is enabled in your Developer org automatically and includes toxicity detection in LLM responses by default. For Makana's use case, we can continue with the default settings. But

if you want to make changes, individual entities or categories can be turned off or on from this page. It can take up to a few minutes for the changes to take effect.

Note: *Pattern-based and field-based data masking for large language models (LLMs) is disabled for Agentforce. Einstein Trust Layer includes several policies and features to help protect sensitive data from misuse or leaks beyond data masking. All information sent to an LLM outside of the Salesforce trust boundary is subject to our zero data retention contract with the LLM provider. Information sent to the LLM is not retained, viewed, or used for training by the provider after the generated response has been sent back to Salesforce. Learn more in Data Masking Limitations in Agentforce.*

Enable Agentforce

Turn on Agentforce to start building, customizing, testing, and launching AI agents.

1. From Setup, in the Quick Find box, enter `Agent`, and then select **Agentforce Agents**.
   If you don't see Agents in Setup, verify that Einstein Generative AI is enabled for your org.
2. To enable Agentforce, turn on **Agentforce**.

?

After you enable Agentforce, refresh the page to see the New Agent button. The New Agent button launches the guided setup for creating an agent. If you don't see the New Agent button, verify your permissions.

Create Your Agent

Use the default Agentforce Employee Agent template to get a head start on creating an agent that assists Makana sales reps by creating opportunities, identifying additional employees who should be involved in the opportunity, and creating tasks for follow-up.

3. From the Agentforce Agents Setup page, click **New Agent.**
4. Select the **Agentforce Employee Agent** template, and then click **Next**.
   The Agentforce Employee Agent template is the default template for the Agentforce Employee Agent type and our general template for assisting your employees.
5. Remove the General FAQ topic, and then click **Next**. For our use case, we'll build a custom topic after we create the agent.

Note: *You add and remove topics from your agent as part of the Agentforce Builder guided setup, but you can always add or remove topics later. To explore all of the standard topics and actions available to add to your agent, from Setup, in the Quick Find box, enter Agentforce Assets, and then select* ***Agentforce Assets****.*

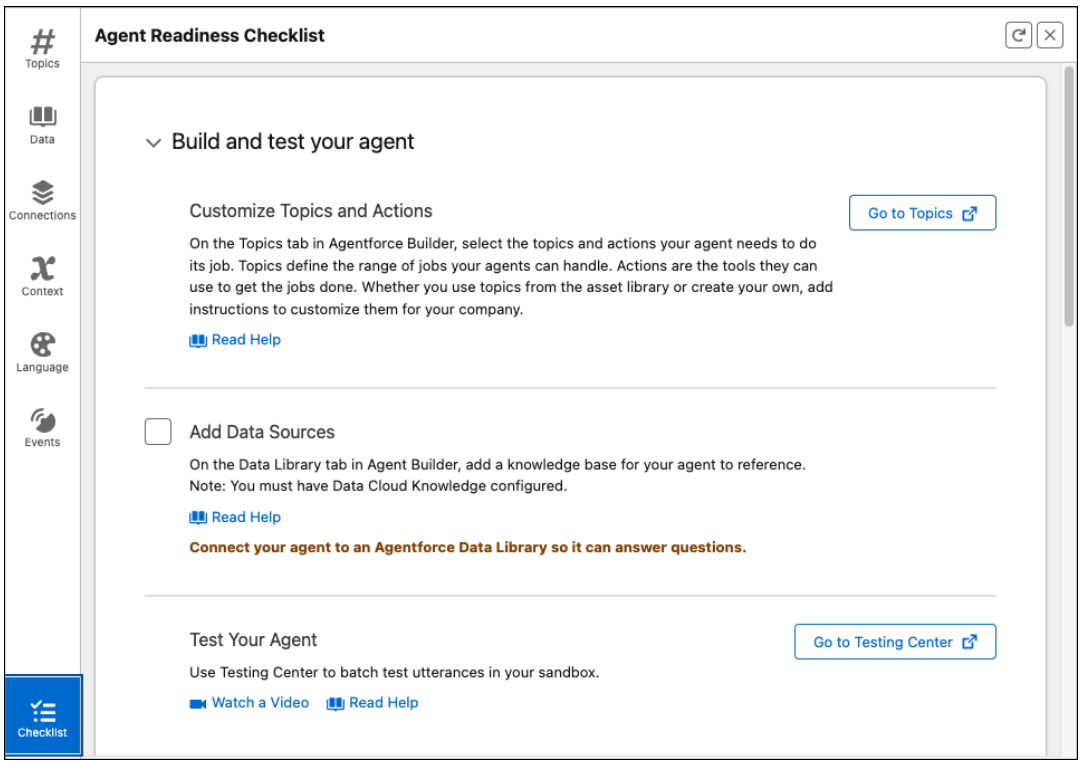
6. Customize your agent with the following fields.
   a. Name: `Makana Opportunity Agent`
   b. API Name: `Makana_Opportunity_Agent`

   c. Description: `Helps sales reps create, collaborate on, and close opportunities.`
   d. Role: `You are an employee assistant whose job is to help sales reps manage their pipeline by creating and managing opportunities.`
   e. Company: `Your company is a global medical device company who manufactures and sells diabetes devices like continuous glucose monitors.`

   The description field is for your own use and helps you and other agent builders in your org identify your agent. The role and company fields are for use by your agent and give your agent important context for the kinds of tasks it can handle and how to respond appropriately to Makana's employees.
7. Select **Keep a record of conversations with enhanced event logs to review agent behavior**. This setting adds customer messages to agent event logs, which capture all session activity to help you monitor and troubleshoot your agent. Learn more about agent event logs.
8. Click **Next**.
9. In the Select data sources step, check the Data Library field. If the field is empty, you can skip this step. If it contains a data library, remove it. For Makana's Employee agent use case, we don't need to ground the agent in any company data. You can always create a data library and add it to your agent later. Learn more about data libraries.
10. Click **Create**.

After your agent is created, it's opened in Agentforce Builder. Leave your agent open in the builder. We'll come back to make changes after tackling your agent's system messages.

Note: *In Agentforce Builder, use the Agent Readiness Checklist to help you configure and deploy a fully functional Agentforce Employee agent. The checklist tracks your progress against agent setup tasks and lets you know when a step is incomplete or otherwise requires your attention.*



Customize the Agent's System Messages

Now that you've created a basic agent, let's get your agent ready for its first conversation by defining system messages. These messages introduce your agent and handle major errors.

1. You should already have the Agentforce Agents page open in a separate tab. Refresh the page to see your new agent.
   If you don't have the Agentforce Agents page open, from Setup, in the Quick Find box, enter Agent, and then select **Agentforce Agents**.
2. Click the name of your agent to open your agent's details page.
3. Select the **System Messages** tab.
4. In the Welcome Message field, enter `Hi, I'm Makana's AI sales opportunity assistant. I can create opportunities and tasks, and I can connect you with colleagues who can help you close and win. I may make mistakes, so be sure to check my work. How can I help?`
   This message is sent at the start of every agent conversation. Makana makes sure to introduce the agent as an AI agent and not a human, which helps establish customer transparency and trust (and is a legal requirement in some regions). Makana also knows that their agent, like all other generative AI applications, can make mistakes. So it encourages its users to carefully review agent responses for accuracy.
5. Verify that the error message is `Something went wrong. Try again.`
   Makana keeps this error message generic because it applies only in the case of non-recoverable system errors, such as a lost connection or LLM outage. The agent

attempts to handle more minor, recoverable errors naturally as part of the agent conversation.

6. Under Conversation Recommendations on the Welcome Screen, leave the setting enabled. In Agentforce Employee agents connected to Lightning Experience and the Salesforce mobile app, conversation recommendations are clickable suggested requests or questions that make it easier for users to chat with your agent. Welcome recommendations appear in the Agentforce panel at the start of an agent conversation and are dynamically generated based on the user's page context and the topics assigned to an agent.

Note: *In Agentforce Employee agents connected to Lightning Experience and the Salesforce mobile app, contextually relevant conversation recommendations also appear after agent responses. These are enabled by default and can be turned on or off from the Details tab of your agent's details page.*

7. Save your changes.

Learn More: *Makana's agent is English only, so we don't have to adjust any language settings. But if you want to change the language or add more languages to an Agentforce Employee agent, this is a good time in the process to get started. Learn more in Salesforce Help.*

- *Update Language Settings*
- *Considerations for Agentforce Employee Agent*
- *Use Additional Languages in Enhanced Chat Conversations*
  In this guide, Makana isn't connecting their agent to Enhanced Chat, so these instructions for using additional languages don't apply. But you can add an Agentforce Employee agent to an Experience Cloud site with Enhanced Web Chat.

Configure the Agent with Topics and Actions

Now that you've created the agent, it's time to give it the topics and actions it needs to help Makana's sales reps create opportunities and relate them to the relevant account, identify other roles in the company that should be involved in the opportunity, and automatically create related tasks. To do this, you'll build a single custom topic, Opportunity Creation, that contains three agent actions.

- A custom flow-based action called Create Opportunity that collects information from the seller, creates the opportunity record, and links it to the account
- A custom prompt template-based action called Identify Recommended Sales Team that identifies relevant specialists in other roles at Makana that can help with the sale and provides a reason for why they should be involved
- The standard Create a To-Do action, which creates tasks for follow-up or assignments to complete

Topic Fields

All topics contain a name, classification description, scope, and instructions.

Field	What It Is	How an Agent Uses It
Name	The API name of the topic. 2–3 words that describe what the topic does in plain language.	An agent uses the topic name to determine when to use a topic in a conversation. The agent compares the topic name and classification description to the user's question or request, including the recent conversation history. Then the agent selects the best match.
Classification Description	1–3 sentences that describe what a topic does and the types of user requests that should be classified into this topic.	An agent uses the classification description to determine when to use a topic in a conversation. The agent compares the topic name and classification description to the user's question or request, including the recent conversation history. Then the agent selects the best match.
Scope	A specific job description that sets boundaries for your agent.	After an agent selects a topic, the scope tells the agent what it can and can't do with a topic.
Instructions	Guidelines that tell your agent how to handle conversations in the context of the topic, your business case, and the conversation in general. Each instruction is a single topic-specific guideline.	After an agent selects a topic, it uses topic instructions, along with the names and instructions of the associated actions, to decide what actions to run, the sequence of steps the agent should follow, and how the agent should format its responses. The agent also uses topic instructions to understand how to handle exceptions, such as when to ask for more information or when to transfer the conversation to a representative.

Learn More: *Topics are fundamental to defining your agent's behavior. Want to go deeper? Learn more in Salesforce Help.*

- *Agent Topics*
- *Best Practices for Topic Instructions*

Agent Actions
In addition to these fields, a topic contains actions that are relevant to the topic's use case. Each action in a topic is a task that your agent can perform when that topic is selected in the conversation.

An agent action includes:

- A natural language name and instructions that tell the agent how and when to use the action, how to retrieve required inputs, and how to format and use outputs.
- The Salesforce functionality that the agent action calls to get information or perform a task, called a reference action. For example, an agent action can call an API to retrieve data, a flow to update a record, a prompt template to generate a response, an Apex class to run custom business logic, or a predictive model to make a recommendation.

Create a Custom Opportunity Creation Topic and Add the Identify Record by Name Standard Action

Create a custom Opportunity Creation topic, which you'll add all of your agent actions to.

Note: *These steps create the Custom Opportunity Topic in your agent in Agentforce Builder. The topic is automatically added to your agent, but it's available only to this agent and agent version. If you'd rather create it as a topic that's available to multiple agents and versions, create it from the asset library. From Setup, in the Quick Find box, enter* `Agentforce Assets`*, and then select* ***Agentforce Assets****. Then click* ***New****. Once your topic is available in the asset library, you can add it to your agent in Agentforce Builder.*

1. You should still have your agent open in Agentforce Builder in a separate tab.
   If you don't have your agent open, from the Agentforce Agents Setup page, click the name of your agent and then click **Open in Builder**. You should be in the Topics tab.
2. Click **New**, and then select **New Topic**.
3. Click **Next** to skip the optional What do you want this topic to do? field.
   If you want to use AI to generate your topic, you can enter a natural language description of what you want your agent to be able to do with this topic here. It's a great way to get a head start on all of the topic fields, including basic topic instructions. For this use case, you're building an agent that we've validated, so you can skip this step.
4. Complete the following fields.

Field	Value
Name	Opportunity Creation
API Name	Opportunity_Creation

Classification Description	Help users create an opportunity, connect with internal resources, and identify and create tasks for each step.
Scope	Your job is only to help users create opportunities by gathering the necessary information, determining whether internal resources should be engaged, and creating tasks as needed.
Instruction	Before creating an opportunity, check whether the user is currently on an account record (currentRecordId). If the user isn't on an account record or if the currentRecordId context variable is empty, ask the user what account they want to create an opportunity for and use the Identify Record by Name action to look up the account. Always try to determine the account from context. Never ask a user for an account or run the Identify Record by Name action if you can determine the account from context.
Instruction	Use the Create an Opportunity action to create an opportunity. You must return the new opportunity record in the conversation before running any additional actions.
Instruction	Collect opportunity details from the user. If the user doesn't specify a close date, set the close date to 30 days from today.
Instruction	After you show the user the opportunity you created, ask the user if they want you to recommend internal team members who can help them close the deal. If they say yes, run the Identify Recommended Internal Team action. Always show the user the output of the Identify Recommended Internal Team action.
Instruction	For each recommended internal team member, automatically create a task for the user to connect with them. Show the tasks in the conversation as hyperlinks to the task records you create.

5. Bonus: Use AI to evaluate our new instructions. On any instruction, click ✦ and see whether Agentforce suggests any improvements. Then click **Next**.
6. From the list of standard agent actions, select **Identify Record by Name**, and then click **Finish**. The Opportunity Creation topic is added to your agent and is available only in this agent and agent version.

Standard Agent Action	What It Does

Identify Record by Name	Uses a standard core action to searches for Salesforce records by name and return a list of matching record IDs

The agent creates an opportunity based on the account record page the user is viewing. If the user isn't on an account record page (for example, when the user is accessing the agent via Slack), the agent will use the Identify Record by Name action to look up the account the user wants to create an opportunity for.

Create a Custom Create an Opportunity Action

Now you'll build the first custom action, which collects information from the seller, creates the opportunity record, and links it to the account. For this action, first you'll build the Create an Opportunity flow. Then you'll use the flow to create the custom Create an Opportunity agent action.

Create the Create an Opportunity Flow

Plan and build the Create an Opportunity to flow to use with a custom agent action.

Before you create the flow, open an opportunity from the Sales app to review available fields and make decisions. This is a great way to plan the inputs and outputs for your flow. For example, you can require the user to provide the details for some or all of the fields, you can configure the agent to make decisions for you, or you can leave some fields blank or set to default values. Consider the user experience in the agent conversation–what information does it make sense to collect conversationally, and how can your agent best add value to your business?

For Makana's use case and for their first iteration of their agent, you'll keep things relatively simple and handle mostly only required opportunity fields. In the flow, you'll configure the agent to ask the user to provide the project name, close date, description, and a next step and create an opportunity based on the user's response.

1. From Setup, in the Quick Find box, enter `Flows`, and then select **Flows**.
2. Click **New Flow**.
3. To create an autolaunched flow from scratch, select the **Autolaunched** category, and then select **Autolaunched Flow (No Trigger)**.
4. On the Manager tab of the toolbox panel, create these resources.

   accountId: The agent uses the account ID of the account record that the user is currently on to add the account name to the Account Name field on the opportunity and link the opportunity to the account. If the user isn't looking at an account or the agent can't determine the account from context, the agent asks the user for an account and uses the Identify Record by Name action to look up the account ID.

Field	Value
Resource Type	Variable

API Name (case-sensitive)	accountId
Description	The recordId or accountId of the account record the user is creating an opportunity for. The accountId always starts with “001.” Alternatively, search for the record by name for this value.
Data Type	Text
Availability Outside the Flow	Available for input

closeDate: The agent asks the user for this information and adds it to the Close Date field on the opportunity. If the user doesn’t provide it, the agent defaults the close date to 30 days from the opportunity creation date.

Field	Value
Resource Type	Variable
API Name (case-sensitive)	closeDate
Description	The date that the deal is expected to close. If the user doesn’t offer this date, then assume 30 days from today.
Data Type	Date
Availability Outside the Flow	Available for input

nextStep: The agent asks the user for this information, formats it, and adds it to the Next Step field on the opportunity.

Field	Value
Resource Type	Variable

API Name (case-sensitive)	nextStep
Description	The next step for the user to take, provided by the user. Include a due date. Format as [MM/DD]: Details of task.
Data Type	Text
Availability Outside the Flow	Available for input

description: The agent asks the user for this information, formats it in a bulleted list, and adds it to the Description field on the opportunity.

Field	Value
Resource Type	Variable
API Name (case-sensitive)	description
Description	The description of the opportunity in detail, provided by the user. Format in a bulleted list.
Data Type	Text
Availability Outside the Flow	Available for input

projectName: The agent asks the user for this information and adds it to the Name field on the opportunity.

Field	Value
Resource Type	Variable
API Name (case-sensitive)	projectName
Description	The name of the project that is creating this opportunity. For example: Medical

	Supplier, Direct-to-Consumer Sales
Data Type	Text
Availability Outside the Flow	Available for input

opportunityCreated: Stores the opportunity created by the flow.

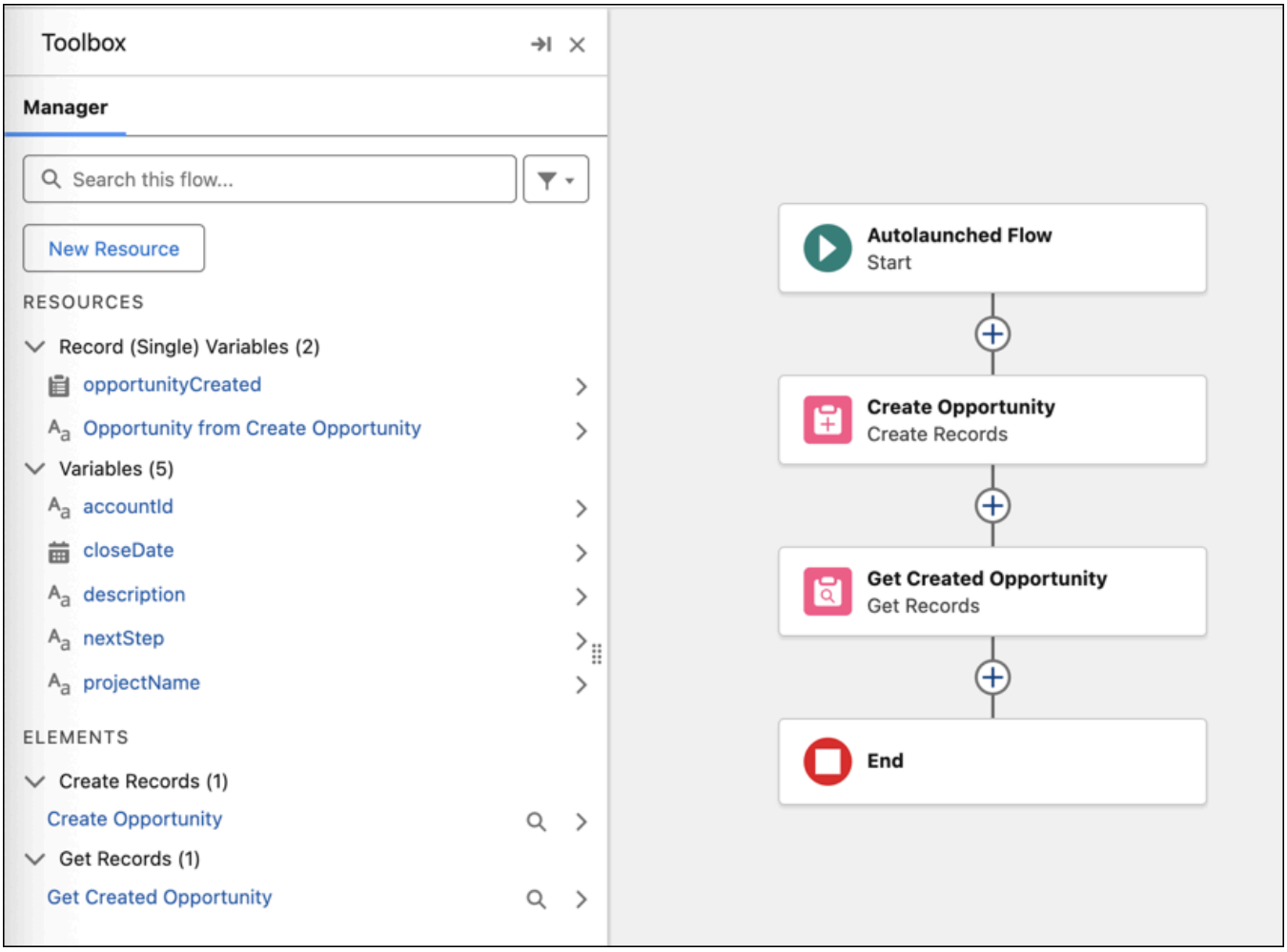
Field	Value
Resource Type	Variable
API Name (case-sensitive)	opportunityCreated
Description	The opportunity that was created.
Data Type	Record
Object	Opportunity
Availability Outside the Flow	Available for output

5. Add a Create Records element. This element creates the opportunity for us, based on the account record and the information provided by the user.
    a. For the label, enter `Create Opportunity`.
    b. For the description, enter `Creates an opportunity associated with an account record, based information provided by the user.`
    c. For how to set record field values, select **Manually**.
    d. For the object, select **Opportunity**.
    e. Under Set Field Values for the Opportunity, map the following fields to the variable resources you created. You also set the Stage field value to Qualification by default. To add fields, click **Add Field**.

Field	Value
Close Date	closeDate
Name	projectName
Stage	Qualification
Account ID	accountID
Description	description

Next Step	nextStep

6. Add a Get Records element to make sure the agent knows how to retrieve the opportunity it created and pass it back into the agent conversation for the user to review.
    a. For the label, enter `Get Created Opportunity`.
    b. For the description, enter `Retrieves the opportunity created by the Create Opportunity Create Records action.`
    c. For the data source, select **Salesforce Object**. Then for the object, select **Opportunity**.
    d. To make sure the flow returns the right opportunity, filter opportunity records.
        i. For condition requirements, select **All Conditions Are Met (AND)**.
        ii. For the field, select **Opportunity ID**.
        iii. For the operator, select **Equals**.
        iv. For the value, select **Opportunity from Create Opportunity**.
    e. Specify the opportunity fields to show in the agent conversation.
        i. For How to Store Record Values, select **Choose fields and assign variables (advanced)**.
        ii. For Where to Store Field Values, select **Together in a record variable**.
        iii. Under Select Variable to Store Opportunity, for the record, select **opportunityCreated**.
        iv. Under Select Opportunity Fields to Store in Variable, search for and select the **Id**, **Name**, and **NextStep** fields. To add a field, click **Add Field**.
7. Save the flow.
    a. For the flow label, enter `Create Opportunity.`
    b. For the description, enter `Create an opportunity for the account the user is working on.`
8. Activate the flow.



Note: *We've built and tested this flow ourselves, so we're confident that it works as expected. But it's always a best practice to debug your flow to verify how it works before you activate it. To debug the flow, you'll need to provide values for the accountId, closeDate, description, nextStep, and projectName variables. For this simple debugging test, you can use the account ID from any account in your org. From the App Launcher, open the Sales app and open an account. Then you can grab the ID for the account from the URL–it's an 18-digit ID beginning with "001" and located between the last two backslashes of the URL. For example, 001gK000002Nxxxxxx.*

If you created a Developer org to complete this guide, the prepopulated accounts in your org aren't necessarily representative of Makana customers. For this test, that's okay–it's still worth verifying that your flow works as expected.

Create a Custom Agent Action Based on the Create Opportunity Flow

Next, create the Create an Opportunity custom agent action, so you can use the Create an Opportunity flow with your agent.

Note: *These steps create the Create Opportunity action in your agent in Agentforce Builder. The action is automatically added to your topic and agent, but it's available only to this agent and agent version.* ***If you'd rather create it as an action that's available to multiple agents and versions, create it from the asset library. From Setup, in the Quick Find box, enter*** `Agentforce Assets`, ***and then select*** ***Agentforce Assets****. Then click* ***New****. Once your action is available in the asset library, you can add it to your agent in Agentforce Builder.*

1. Back in Agentforce Builder, from the Topics tab, click the **Opportunity Creation** topic to open it.
2. Click the **This Topic's Actions** tab.
3. Click **New**, and then select **Create New Action**.
4. For the reference action type, select **Flow**.
5. For the reference action, select your **Create Opportunity** flow. Then click **Next**.
   If you don't see the flow, refresh the page and try again.
6. The action instructions autopopulate from the flow, so no change is needed.
7. For the loading text, enter `Creating your opportunity…`
8. For each input, select **Require input**. The input instructions autopopulate from the flow. We wrote the flow descriptions with the agent in mind, so no further change is needed.
9. For the output, select **Show in conversation**. The output instructions autopopulate from the flow. We wrote the flow descriptions with the agent in mind, so no further change is needed.
10. Click **Finish**. Your new action is added to the Opportunity Creation topic and is available only in this agent and agent version.

Create a Custom Identify Recommended Internal Team Action

Build your second custom action, which identifies relevant specialists in other roles at Makana that can help with the sale and provides a reason for why they should be involved. For this action, first you'll create the Identify Recommended Internal Team prompt template. Then you'll use the prompt template to create the custom Identify Recommended Internal Team agent action.

Create the Identify Recommended Internal Team Prompt Template

First, create a prompt template that identifies internal specialists that can help sales reps with their opportunity. You'll add prompt instructions and enrich the prompt with resources from the opportunity record.

1. From Setup, in the Quick Find box, enter Prompt Builder, and then select **Prompt Builder**.
2. Click **New Prompt Template**.
3. For prompt template type, select **Flex**.
   Flex templates allow you to select your own inputs, whereas the inputs for other prompt template types are predefined.
4. Enter prompt template details.
   a. For the prompt template name, enter `Identify Recommended Internal Team`.
   b. For the API name, enter `Identify_Recommended_Internal_Team`.

   c. For the description, enter `For an opportunity, help the seller identify who they might need to be included on the opportunity.`
5. Under Inputs, click **Add**, and then add the Opportunity object.
   When you add the Opportunity object as an input, you can add merge fields to your prompt instructions to include data from opportunity fields or related lists, without having to build any additional automations.
   a. For name, enter `Opportunity`.
   b. For API name, enter `Opportunity`.
   c. For source type, select **Object**.
   d. For object, search for and select **Opportunity**.
6. Click **Next**, and you're redirected to the Prompt Template Workspace. The Prompt Template Workspace is where you add the details for your prompt template.
7. Copy and paste the following prompt instructions into your prompt template.
   The prompt instructions include the agent's objective, output, and the contextual information required to generate the output, including relevant fields and a related list from the opportunity record and descriptions of all of the internal roles at Makana available to help a sales rep with the opportunity.

```
OBJECTIVE:
You are a sales operations manager. Identify who from our
internal team should be supporting this opportunity.

OUTPUT:
Output the two most relevant roles to the opportunity and a
reason why in your response. Make sure the reason is concise
(less than 300 characters), specific to the opportunity, and
includes how the role can help close the deal.

CONTEXT:
Here is the opportunity:
Description: {!$Input:Opportunity.Description}
Next steps: {!$Input:Opportunity.NextStep}
Customer contacts involved:
{!$RelatedList:Opportunity.OpportunityContactRoles.Records}

Here are the roles we have:
Clinical applications specialist
Clinical applications specialists (CAS) implement, support, and
train Makana customers' on CGM3000 and CGM3000 Smart systems
and devices. They help train patients and healthcare
professionals to use our glucose monitoring devices in
professional and at-home settings, customize our monitoring
software to meet patient and professional needs, and
troubleshoot our glucose monitoring devices and systems. They
can help build customer confidence in products and trust in
ongoing support.
```

```
Data analyst
Data analysts provide data insights that can help close a sale
and identify growth opportunities. They can analyze current
Makana customer purchase data and sales interactions to make
recommendations for relevant products, services, or quantities,
as well as track inventory. They can create reports and
dashboards for internal or external use to demonstrate
potential or realized business value, ROI, and improved health
outcomes.

Product marketing manager
Product marketing can provide a deeper dive into market
research, clinical trial data, and case studies that highlight
the Makana CGM3000 and CGM3000 Smart's effectiveness. They can
also help sales reps develop highly specific, evidence-backed
presentations tailored to their prospect's needs.

Compliance specialist
Compliance specialists can help address a prospect's concerns
related to FDA approvals, safety standards, and other
regulatory hurdles. Compliance specialists can explain Makana's
system for monitoring CGM3000 device performance and reporting
adverse events. They can review and approve all promotional
materials to ensure compliance with FDA and other regulatory
approvals.  If necessary, they can train junior sales reps on
ethical selling practices, data privacy laws (HIPAA), and
regulations concerning interactions with healthcare
professionals.
```

8. Save your prompt template.
9. Preview a generated response for your prompt template.
    a. Click **Preview Settings**.
    b. Under Inputs, search for and select an opportunity record.

Note: *If you created a Developer org to complete this guide, the prepopulated opportunities in your org aren't necessarily representative of Makana customers. For this test, that's okay–it's still worth previewing responses based on one or more opportunities to validate the prompt instructions.*

    c. Under Outputs, make sure **Generate Response** is turned on.
    d. Click **Preview.**
    e. Review the output and make any adjustments to the prompt instructions.

Note: *To ensure that your prompt template yields consistent results, generate multiple responses per record without making any changes to the prompt instructions, and then compare the responses.*

10. When you're satisfied with your response, activate your prompt template.

Create a Custom Agent Action Based on the Identify Recommended Internal Team Prompt Template

Next, create the Identify Recommended Internal Team custom agent action, so you can use the Identify Recommended Internal Team prompt template with your agent.

Note: *These steps create the Identify Recommended Internal Team action in your agent in Agentforce Builder. The action is automatically added to your topic and agent, but it's available only to this agent and agent version. If you'd rather create it as an action that's available to multiple agents and versions, create it from the asset library. From Setup, in the Quick Find box, enter* `Agentforce Assets`*, and then select* ***Agentforce Assets****. Then click* ***New****. Once your action is available in the asset library, you can add it to your agent in Agentforce Builder.*

1. Back in Agentforce Builder, from the Topics tab, click the **Opportunity Creation** topic to open it.
2. Click the **This Topic's Actions** tab.
3. Click **New**, and then select **Create New Action**.
4. For the reference action type, select **Prompt Template**.
5. For the reference action, select your **Identify Recommended Internal Team** prompt template. Then click **Next**.
   If you don't see the prompt template, refresh the page and try again.
6. The action instructions autopopulate from the prompt template, so no change is needed.
7. For the loading text, enter `Assembling the right team...`
8. Customize the Opportunity input.
   a. For the input instructions, enter `The opportunity the user is either looking at or that was just created.`
   b. Verify that **Require input** is selected.
9. Customize the Prompt Response output.
   a. For the instructions, enter `Return the two most relevant roles to the opportunity and how they can help close the deal.`
   b. Leave Show in conversation deselected. This allows the agent to determine how to format the response in the agent conversation.
10. Click **Finish**. Your new action is added to the Opportunity Creation topic and is available only in this agent and agent version.

Create a Custom Create Tasks to Connect Action

Build your third custom action, which creates tasks to connect with the specialists identified by the Identify Recommended Internal Team action and automatically relates the tasks to the opportunity created by the Create Opportunity agent action. For this action, first you'll create the Create Tasks to connect flow. Then you'll use the flow to create the custom Create Tasks to Connect agent action.

Note: *If you created a free Developer org to complete this guide, create the Create Tasks to Connect action from scratch. But if you're using a different kind of org, you may have access to the Create a To Do standard agent action. This action uses a standard core action to create tasks for sellers, including follow-ups and assignments to complete. If you have access to it, you can add the Create a To Do action to your agent from the asset library and skip the steps to create a custom Create Tasks to Connect action.*

Create the Create Tasks to Connect Flow

Build the Create Tasks to Connect flow to use with a custom agent action.

1. From Setup, in the Quick Find box, enter `Flows`, and then select **Flows**.
2. Click **New Flow**.
3. To create an autolaunched flow from scratch, select the **Autolaunched** category, and then select **Autolaunched Flow (No Trigger)**.
4. On the Manager tab of the toolbox panel, create these resources.

   createdOpportunityId: The agent uses this ID to relate newly created tasks to the opportunity previously created by the Create Opportunity action.

Field	Value
Resource Type	Variable
API Name (case-sensitive)	createdOpportunityId
Description	The ID of the opportunity created by the Create Opportunity action.
Data Type	Text
Availability Outside the Flow	Available for input

   createdTaskId: Stores the task created by the flow.

Field	Value
Resource Type	Variable
API Name (case-sensitive)	createdTaskId
Description	The ID of the newly created task.

Data Type	Text
Availability Outside the Flow	Available for output

descriptionOfTeamMember: The agent populates the description of the newly created tasks with the reasoning for why the team member was recommended to help close the opportunity.

Field	Value
Resource Type	Variable
API Name (case-sensitive)	descriptionOfTeamMember
Description	A summary of why the internal team member recommended by the Identify Recommend Internal Team action will be helpful in closing the opportunity.
Data Type	Text
Availability Outside the Flow	Available for input

dueDate: The agent sets the due date for the task to one week from the date the task is created.

Field	Value
Resource Type	Variable
API Name (case-sensitive)	dueDate
Description	Set the due date to one week from today.
Data Type	Date
Availability Outside the Flow	Available for input

recommendedTeamMember: The agent populates the subject of the newly created task with the team member recommended to help close the opportunity.

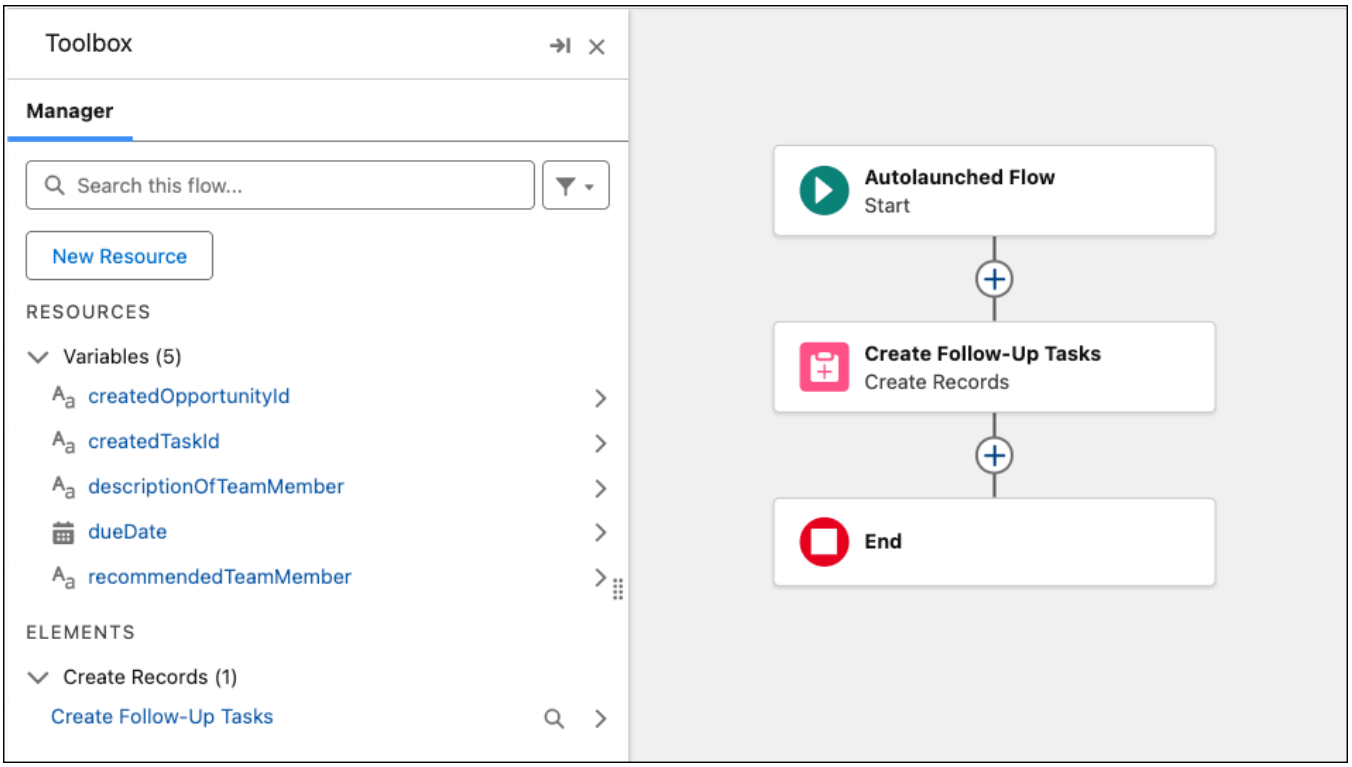
Field	Value
Resource Type	Variable
API Name (case-sensitive)	recommendedTeamMember
Description	One of the internal roles recommended by the Identify Recommended Internal Team action. Format of the subject field should be Connect with [Internal Team Member Role]. For example: Connect with Product Marketing Manager
Data Type	Text
Availability Outside the Flow	Available for input

5. Add a Create Records element. This element creates the tasks for us, based on the output of the Identify Recommended Internal Team action.
    a. For the label, enter `Create Follow-Up Tasks.`
    b. For the description, enter `Creates a task based on the output of the Identify Recommended Internal Team action.`
    c. For how to set record field values, select **Manually**.
    d. For the object, select **Task**.
    e. Under Set Field Values for the Task, map the following fields to system variables and the variable resources you created. You also set the default status of the task to Not Started. To add fields, click **Add Field**.

Field	Value
Assigned To ID	Running User > Id
Due Date Only	dueDate
Description	descriptionOfTeamMember
Related To ID	createdOpportunityId

Status	Not Started
Subject	recommendedTeamMember

6. Select **Manually assign variables (advanced)**.
7. Under Store Task ID in Variable, for the Variable field, select the **CreatedTaskId** variable.
8. Save the flow.
    a. For the flow label, enter `Create Tasks to Connect.`
    b. For the description, enter `Creates a task to connect with recommended internal team members based on the team members identified by the Identify Recommended Internal Team action.`
9. Activate the flow.



Create a Custom Agent Action Based on the Create Tasks to Connect Flow

Next, create the Create Tasks to Connect custom agent action, so you can use the Create Tasks to Connect flow with your agent.

Note: *These steps create the Create Tasks to Connect action in your agent in Agentforce Builder. The action is automatically added to your topic and agent, but it's available only to this agent and agent version. If you'd rather create it as an action that's available to multiple agents and versions, create it from the asset library. From Setup, in the Quick Find box, enter* `Agentforce Assets`*, and then select* ***Agentforce Assets****. Then click* ***New****. Once your action is available in the asset library, you can add it to your agent in Agentforce Builder.*

1. Back in Agentforce Builder, from the Topics tab, click the **Opportunity Creation** topic to open it.
2. Click the **This Topic's Actions** tab.
3. Click **New**, and then select **Create New Action**.
4. For the reference action type, select **Flow**.
5. For the reference action, select your **Create Tasks to Connect** flow. Then click **Next**. If you don't see the flow, refresh the page and try again.
6. The action instructions autopopulate from the flow, so no change is needed.
7. For the loading text, enter `Creating follow-up tasks...`
8. For the createdOpportunityId, descriptionOfTeamMember, and recommendedTeamMember inputs, select **Require input**.
9. For each input and output, the instructions autopopulate from the flow. We wrote the flow descriptions with the agent in mind, so no further change is needed.
10. Click **Finish**. Your new action is added to the Opportunity Creation topic and is available only in this agent and agent version.

Verify Your Changes

Now that you've added all of your actions to your agent, start a preview conversation to make sure the agent performs as expected. You'll learn much more about testing your agent later in this guide, but building and testing your agent is an iterative process of making one or two small adjustments at a time and checking your results.

1. To make sure your preview conversation reflects all of your changes, refresh Agentforce Builder.
2. In the conversation preview panel, click  to add preview conditions to your conversation. Preview conditions let you simulate the context of an agent conversation from the Builder. For Makana's Opportunity Agent, we expect sales reps to be working in the Sales app and creating an opportunity from an account record, so add the following conditions.

Field	Value
currentObjectApiName	opportunity
currentAppName	Sales
currentRecordId	Use the account ID from any account in your org. From the App Launcher, open the Sales app and open an account. Then you can grab the ID for the account from the URL–it's an 18-digit ID beginning with "001" and located between the last two backslashes of the URL. For example, 001gK000002Nxxxxxx. ***Note:*** *If you created a Developer org to complete this guide, the prepopulated*

	accounts in your org aren't necessarily representative of Makana customers. For this test, that's okay–it's still worth previewing responses based on one or more accounts to verify your agent's behavior.

3. Apply your changes.
4. Chat with your agent, and monitor its reasoning on the canvas.
    - Send your agent messages that represent the ways you expect sales reps to ask the agent to create an opportunity. For example, "create an opp."
    - Be prepared to give the agent the information it needs to create the opportunity, including a project name, description, next step, and a close date. For example, "The project name is Outpatient. The deal is supposed to close end of this month. This is a potential order for 1,500 devices for a regional chain of clinics. They'd like to see more clinical implementation documentation and any data from our studies on usability and user compliance. Next step is to figure out what to quote tomorrow."
    - Try variations that don't include all of the required information, and see how the agent responds.
5. If necessary, make adjustments to your agent's topic instructions, action instructions, or action input or output instructions. Then repeat the process.

Learn More: *This is a simplified overview of a quick validation check of your agent's behavior. For a much deeper dive into testing approaches, best practices, and scenarios, continue on to the Test Your Agent section of this guide.*

Give Your Users Required Permissions

Make sure your users have access to your agent and can run the actions you've added to your agent.

Note: *If you created a Developer org to complete this guide, you don't necessarily have any users to assign permissions to. We recommend practicing some of these steps on your own admin user record or a new user that you create. We just want you to know how to plan and implement user access when you create an Agentforce Employee agent.*

Give Your Users Access to Your Agent

Access to Agentforce Employee agents is granted using permission sets or profiles. You can add agent access to existing permission sets or profiles from the page for your agent. You can also add agent access to new permission sets or profiles from their respective setup pages, from the Agent Access category. For this step, grant your own user profile agent access from the agent's details page.

1. From Setup, in the Quick Find box, enter `Agents`, and then select **Agentforce Agents**.

2. To open your agent's details page, click **Makana Opportunity Agent**.
3. Select the **Agent Access** tab.
4. Select the **Profiles with Agent Access** tab, and then click **Add**.
5. Select the **System Administrator** profile, and then click **Add**.

When you activate your agent, it's available in the Agentforce panel in Lightning Experience and in the Salesforce mobile app for all users with access. Users with access will see the Agentforce icon  . You can try it out by activating your agent and then, from the App Launcher, opening the Sales app. Just remember to deactivate your agent again before moving on to the next step.

Required Access to Run Agent Actions

Agentforce Employee agents run in the context of the end user. Whatever your user has access to is what the agent they're chatting with has access to. Makana's agent includes agent actions based on prompt templates and flows, so users need the corresponding permissions for the agent to run them, plus access to the objects the actions read and modify.

USER PERMISSIONS NEEDED	
To run prompt templates:	Permission Set: Prompt Template User OR Permission: Execute Prompt Templates
To run flows:	App Permission: Run Flows OR Grant access to individual flows See Limit User Access to Execute Flows
To view accounts:	Read, Create, Edit access to the Account object
To create opportunities:	Read, Create, Edit access to the Opportunity object

Note: *You can test whether a user has all of the right permissions to use your agent by previewing an agent conversation. Just add the user's ID to the preview conditions in the conversation preview panel.*

Learn More: *This agent's configuration doesn't include Apex classes, but they're commonly used for agent actions. Many agents also use Salesforce Knowledge.* *Review the required agent user permissions for these features and more.*

Test Your Agent

Makana Medical Devices is now entering the testing phase of their new Agentforce employee agent. After integrating AI into their Salesforce workflows, this stage focuses on ensuring the agent performs reliably before it goes live.

Because AI agents are non-deterministic, their behavior isn't always predictable. Makana must test across a range of scenarios to confirm consistent, accurate performance. They need to verify that the agent reliably selects the right topics and actions to complete tasks, and that its responses align with Makana's tone, brand identity, and business needs.

This is where manual testing in Agentforce Builder and automated batch testing in Agentforce Testing Center play key roles. Both methods help Makana validate their agent's behavior before deployment and provide the tools for ongoing quality monitoring. As the agent evolves, either through learning from interactions or updates to the underlying models, continuous testing will be essential to maintain performance over time.

We've organized Makana's testing strategy into two phases: 1) planning and preparation, and 2) core testing and iteration.

- Planning and Preparation: Outlines the foundational steps like defining success metrics and strategy, creating realistic test scenarios, and identifying required data.
- Core Testing and Iteration: Walks through the core testing process, from manual validation and batch evaluation to continuous refinement. This helps teams ensure their agents are ready for real-world deployment and long-term success.

Planning & Preparation

Comprehensive testing begins with clearly defining what success looks like for your AI agent, using measurable criteria aligned with your business objectives. Preparation also involves breaking down the agent into core components, like topics and actions, and identifying the most relevant data for evaluation. Finally, this phase focuses on designing structured test scenarios that reflect real-world customer interactions.

Define Objectives and the Testing Strategy

Before testing begins, clearly define what constitutes success for your AI agent.

- Set SMART goals: Establish specific, measurable, attainable, relevant, and time-bound criteria for your agent's performance, directly linked to the business objectives you identified for your agent.
- Deconstruct the agent: Break down the agent into core components such as topics and actions. Testing each part individually helps to pinpoint inefficiencies at their source.
- Link your testing goals to business value: A clear strategy guides the testing process, ensuring agents are thoroughly fine-tuned.

To shape your testing goals, revisit the strategic goals you set during the ideate stage. Makana's agent is focused on helping employees, especially sellers, create opportunities and relate them to the relevant account. The agent also identifies other roles in the company who should be involved in the opportunity and automatically creates related tasks for the seller. They identified some business objectives early in the process and should use those to determine if the agent is on track.

Once you've outlined your objectives, the next step is to create a testing strategy. You can approach this a number of ways, but your strategy should:

1. Identify test scenarios and set evaluation parameters.
2. Run the tests and validate the results.
3. Make adjustments and repeat step 2 until your agent is working as expected.
4. Retest. Always test whenever you introduce something new, but also test on a regular cadence to make sure your agent is still performing the way you intend.

We'll walk through how Makana approaches each of these steps to test their employee agent. Let's get started!

Create Test Scenarios and Identify Data Requirements

A testing scenario is a structured example used to evaluate how well an AI agent, like the Makana Opportunity Agent, performs in a realistic situation. It simulates a real-world interaction and helps you test whether the agent understands the user input correctly, selects the right topic, runs appropriate actions, and responds accurately, completely, and in the right tone. A strong testing scenario should include:

- Scenario description: A short narrative of the situation or employee context, such as an employee who wants to create a new opportunity for an account and get a team together to work on the deal.
- Test utterance: The actual phrase or question the employee might enter in natural language.
- Expected topic: The topic the agent should classify the utterance into based on how the agent is configured.
- Expected action: The specific action the agent should take in response, like "create an opportunity record and link it to a specific account" or "identify other roles in the company who should be involved in the opportunity."
- Expected response: A brand-appropriate reply based on validated information. The agent's wording may vary but as long as the tone feels right and the info is accurate, that's what

really counts. It serves as a guide, not a script. So when you're reviewing, focus on whether the response makes sense and sounds natural, rather than expecting an exact match.
- Utterance variations: The different ways that customers are likely to phrase a question or request. Plan for variations on your utterances, including utterances that vary based on language differences, different emotional states of the customer, and utterances that change the topic of the conversation.
- Multi-turn: It's important to Multi-turn context to test how the agent handles requests inside of continued conversation.

To help you come up with good test scenarios, review the topics assigned to your agent. Look at the classification description and scope fields that describe the capabilities and parameters your agent should operate within and each instruction that directs how the agent performs. Next, write or generate inputs that test against these details to help make sure your agent acts reliably in each scenario. To give you an idea of what that looks like, here's a sample of Makana's inputs.

- Log the new potential sale for Beta Systems.
- Set up the necessary internal review tasks for the $75,000 opportunity.
- I need to assign the appropriate support roles to this new deal.

And while these are a good start, it's also important to think in terms of how a conversation might flow. Users might ask the same question differently, so use paraphrasing to make sure the agent is recognizing the core request. For example:

- Document the Beta Systems deal for $75,000 in Sales Cloud.
- Register the pending sale for the new patient monitoring systems.
- Please log a $75,000 deal in the Qualification stage right away.

Users might also express themselves differently depending on how they feel. Create inputs that cover polite, frustrated, and neutral states, like:

- Neutral: Assign the standard support resources to the Beta Systems opportunity immediately.
- Polite: Could you kindly initiate the follow-up tasks for the Legal and Product Demo resources on the new opportunity?
- Frustrated: I don't have time for this data entry! Just create the opportunity and get the tasks assigned!

You also want to make sure your agent can handle multi-turn conversations and be able follow along when a user shifts topics. Inputs that test for this include multiple phrases, including a change of topic.

Testing Scenario

Scenario	Opportunity Creation and Resource Assignment
Summary	A sales representative has just completed an initial discovery call and needs to quickly create a new opportunity in Salesforce and automatically trigger the assignment of initial follow-up tasks to the internal deal resources (the MVP focus for Opportunity Management). If the agent fails to accurately capture the opportunity details or assign the necessary tasks, the sales rep's productivity goal will not be met
Expected Behavior	
Define how the agent should understand and respond to this scenario. This ensures clarity when evaluating test results.	
Field	**Description**
Expected Topic	Opportunity Management
Expected Action	Create Opportunity & Assign Deal Resources
Expected Response	"Opportunity [OPP-12345] for Acme Corp has been successfully created in Salesforce at the Qualification stage with a value of $50,000. I have also assigned the follow-up tasks to Legal and Product Specialist resources. They should contact you shortly."
Utterance Variations	
Include different tones and reworded inputs to test recognition of intent and emotional range.	
Variation Type	Test Utterance
Neutral	"Start a new opportunity for Acme Corp for $50k of CGM-3000 sensors. Target close is late Q4."
Polite	"Could you please help me quickly log a new deal for Acme? It's for $50,000 and needs to be set to the Qualification stage right away."
Frustrated	"I'm swamped! Just create this $50k Acme deal quickly and assign the standard resource tasks... I need to get back to selling."
Rephrased	"Need to create an opportunity for Acme Corp. The deal is $50,000 for patient monitoring instruments."
Rephrased	"Log this new deal in Sales Cloud: Acme, $50k, for the Pro BP-200 line."
Multi-Turn Conversation Flow	
Outline how the conversation should progress over multiple turns, including topic shifts.	

Turn	User Input	Agent Response
1	“Create a new opportunity for Acme Corp for a potential deal of $50,000 for the Pro BP-200. Set the close date for next quarter.”	Confirm creation and resource/task assignment. “Opportunity created and resources assigned: Legal Review and Product Demo Prep tasks are now active. What else can I help you with?”
2	“Where is that Legal Review task located in Salesforce?”	Provide task retrieval details. “The Legal Review task is tied directly to the Acme Corp opportunity record and is currently awaiting assignment by the Legal team.”
3	“What about the Product Demo Prep tasks”	Provide task status and details, adhering to the V1 MVP scope of managing tasks for deal resources. “The Product Demo Prep task is currently assigned to the Product Specialist resource and is set to the 'In Progress' status. It is linked directly to the OPP-12345 opportunity record.”
4 (Topic Shift)	“Perfect, thanks. On a different note, what are the current guidelines for employee sick leave?”	Detects the shift and transitions to a general policy topic. “I can certainly look up policy information. Employee sick leave guidelines state that full-time employees are eligible for 10 paid days annually. Do you need details on submitting a request?”

As you go through this process, keep in mind that a good test includes enough test cases to cover various scenarios and edge cases. It should also have a wide range of inputs, contexts, and variations as well as inputs outside the agent's scope or those that challenge its guardrails. These details will help make sure your agent is thoroughly tested. With this foundation in place, you’re ready to move into core testing and iteration, where you’ll validate performance, identify gaps, and refine the agent’s behavior.

Core Testing & Iteration

Core testing and iteration are essential for validating and refining your AI agents. This process focuses on long-term agent success by using comprehensive testing and continuous refinement to ensure agents are reliable, accurate, and aligned with business objectives both before and after deployment.

We'll outline an iterative approach that begins with detailed manual verification and progresses to automated batch evaluations. Manual testing is most effective as a starting point to validate your agent configuration before deployment, as well as for exploratory testing. Next, batch tests expand your testing scope by creating diverse scenarios that are specific to your organization's data. They offer a comprehensive view of an agent's performance by uncovering issue trends and edge cases often missed by smaller-scale tests. The results from large-scale evaluations support continuous improvement through enhanced troubleshooting and quality monitoring.

Manual Testing and Verification

It's essential when you first create an agent to confirm it works as you planned. Manual testing is the starting point for validating your agent's performance. It offers a detailed, step-by-step evaluation of your agent's interactions and utterances. Throughout the agent lifecycle, it's important to revisit manual testing in Agentforce Builder, especially after making changes to your agent.

When manual testing, you'll use the Conversation Preview panel to chat with your agent, then track the agent's topic selection, action execution, and reasoning on the plan canvas. The plan canvas is designed to give you a clear view into the AI agent's decision-making, so you can see how it chooses topics and actions.

Makana Medical Devices has built their opportunity agent to handle creating opportunities and associating them with accounts, identifying potential teammates to help work on the deal, and setting up tasks related to the opportunity. Before fully rolling out the agent, the sales team wants to check it accurately understands employee's requests and initiates the correct tasks.

1. From the Agentforce Agents Setup page, Makana launches the Makana Opportunity Agent in Agentforce Builder.
2. Using the Conversation Preview panel, a team member enters a typical request designed to test the Opportunity Management workflow: `Log this new deal- project name Equipment Upgrade, $50k, for the Pro BP-200 line.`
3. In the plan canvas, the team checks the agent's topic selection. They see the agent has correctly identified the Opportunity Creation topic for this query, indicating that the classification and topic instructions are functioning properly.
    a. If it had chosen an unrelated topic the team would review and adjust the topic's classification description to better guide the agent.
4. Next, the team observes which actions the agent selects within the Opportunity Creation topic. The agent should start with the Create Opportunity custom flow-based action, followed by the Identify Recommended Sales Team action and the Create a To-Do action.
    a. If it had chosen an unrelated action, the team would adjust the action's instructions or refine when specific actions should be triggered.
5. The team reviews the agent's reasoning in the plan canvas, following how it parsed the user's question and arrived at the topic and action decisions. This step confirms that the AI's decision-making aligns with expectations.

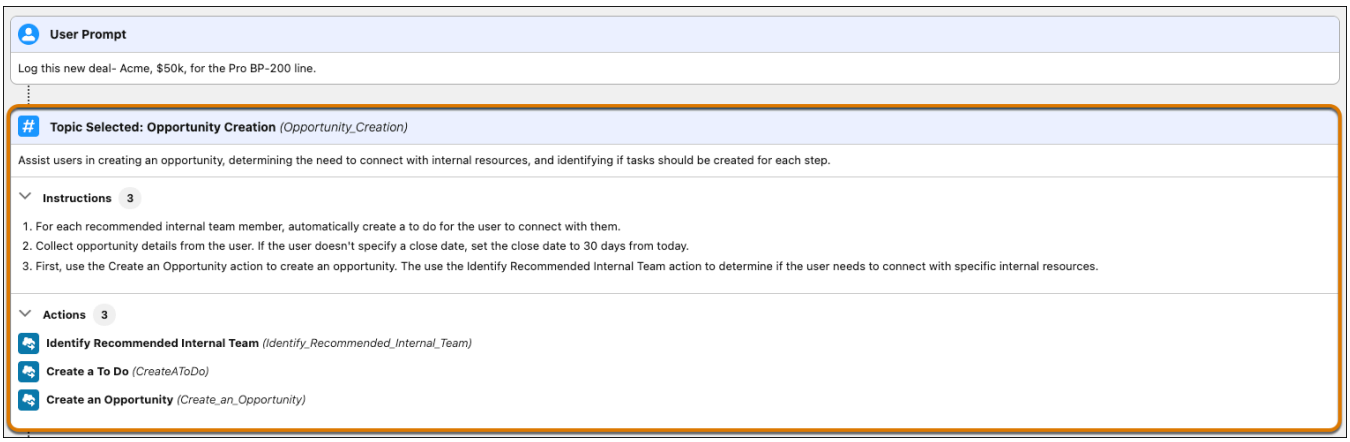
   a. If the agent's reasoning showed any unexpected or missed steps, the team would confirm that the agent correctly uses context and memory, and grounds responses appropriately.
6. After making any necessary adjustments to topics or actions, the team restarts the conversation to test the improved agent responses. This iterative process continues until the service agent consistently delivers accurate and helpful answers.

How did Makana assess agent performance?

The process above shows how Makana assessed their agent's quality through four checks. Let's take a closer look at each one.

Check the Topic Selection

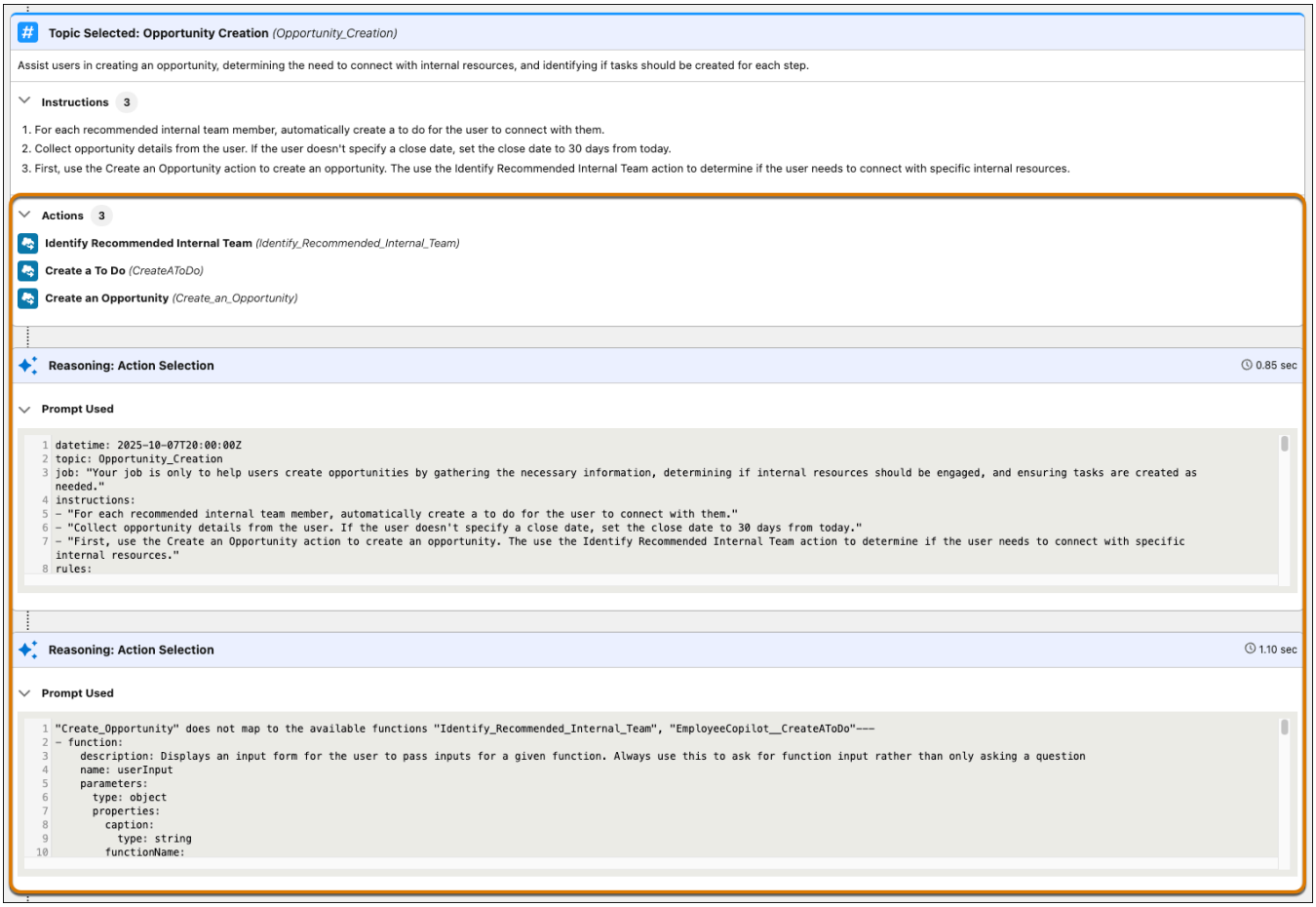
Start by checking the initial utterance and topic selection to verify the agent correctly classified the user's utterance. For the query about creating an opportunity, the agent should select the Opportunity Creation topic. If the agent selected an unexpected topic, that indicates that you should adjust the topic's classification description or instructions. For instance, if the Opportunity Creation topic instructions specify "Only create opportunities for existing accounts," testing the agent's adherence by asking it to create an opportunity for a nonexistent company will ensure the instructions are followed correctly.



To adjust a specific topic's instructions, from the Topics tab, select the topic you want to improve. In Topic Details, under Topic Configuration, you'll see the topics classification description, defined scope, and instructions. You can directly compare the user utterance with the topic instructions.

Check the Action Selection

Next, take a look at the action selection. The Opportunity Creation topic contains three required actions: the custom flow-based Create Opportunity, the custom prompt-based Identify Recommended Sales Team, and the standard Create a To-Do action. If the agent selects an irrelevant action, that indicates that you should adjust the instructions for that action, or that you need to adjust the topic instructions for when and how to use that action.



Check the Agent's Reasoning

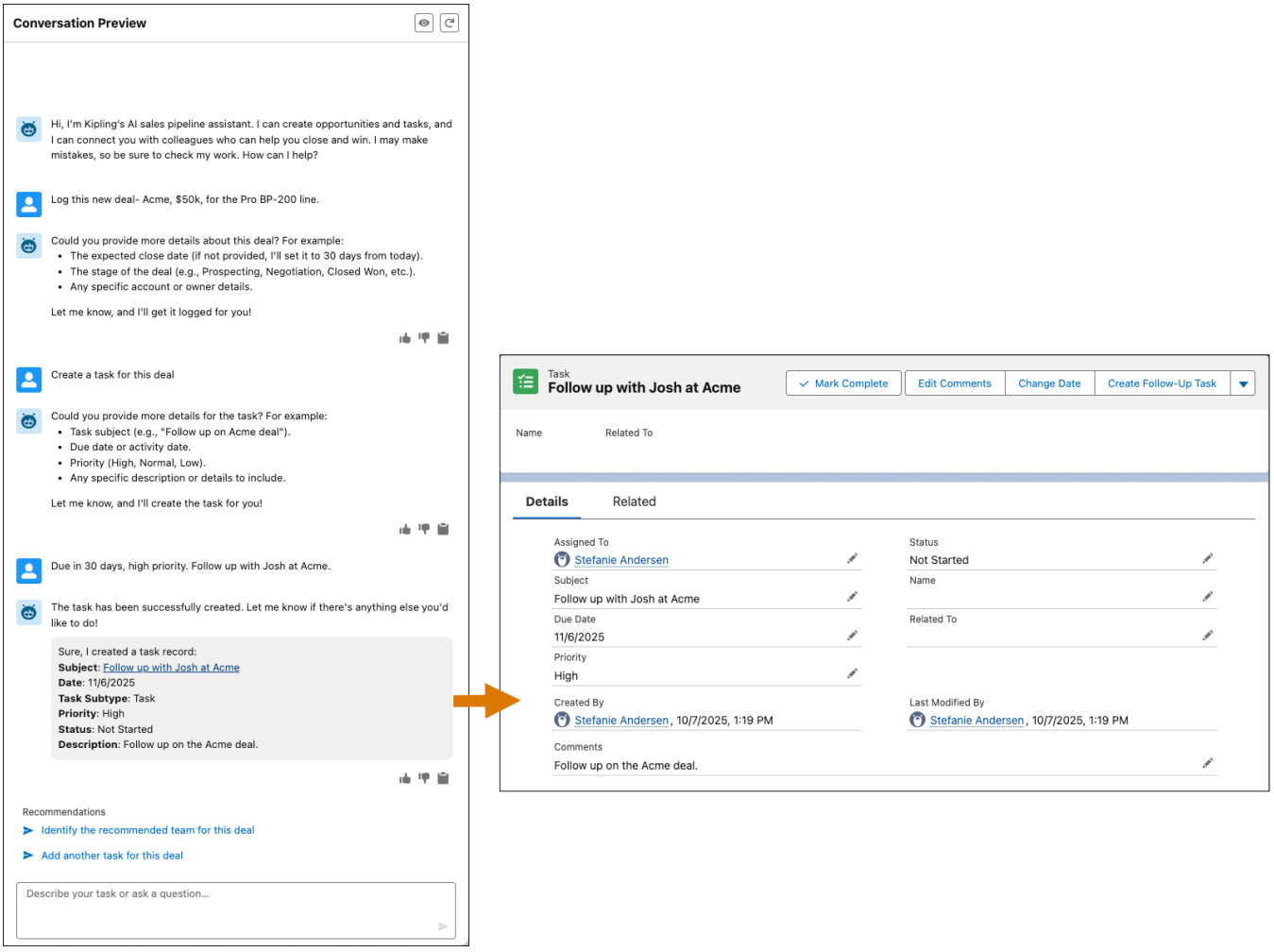
After reviewing the agent's topic and action selection, turn your attention to its reasoning, this is where the agent's "intelligence" is revealed. Use the Plan Canvas to track and validate the agent's decision making, ensuring it makes accurate predictions and consistently follows regulatory rules. The Plan Tracer helps you confirm that the agent correctly uses context and memory, carrying key information like the associated account, throughout the conversation. To prevent hallucinations or non-compliant responses, check the response is adequately grounded to ensure all user-facing information is sourced exclusively from approved, reliable data.

Learn More: *For more information about the reasoning engine, see How Agentforce Works.*

Check Your Other Salesforce Functions

Because each action runs a specific flow, Apex class, or prompt template, issues may stem from these underlying components rather than the agent instructions. After running a test, confirm that the action executed as expected by comparing the results to the action instructions. For example, if the agent chose the Create Opportunity action, you can manually verify that a new opportunity was actually created in Salesforce with all of the correct information populated from the conversation. Or for the Identify Recommended

Sales Team action is a custom prompt template, you could also verify that the output adheres to the expected format.



Batch Testing and Automated Evaluation

Testing individual utterances in Agentforce Builder is a good strategy for targeted testing, topic and action refinement, and troubleshooting. But it can be time consuming if you want to get an overall idea of your agent's reliability across a broader set of real-world scenarios. Take advantage of automation in the testing process to help you efficiently launch reliable and trustworthy agents. In the Testing Center, you can streamline large-scale testing by quickly generating up to 100 realistic test scenarios using AI. Or you can manually draft your own testing scenarios with a CSV template and upload them for testing. Once you've signed off on your tests, you can run up to 10 test jobs with up to 1,000 test cases per test.

Once Makana Medical Devices completes the manual testing of their opportunity agent, they can move on to batch testing to evaluate how well the agent performs at scale. Using the Agentforce Testing Center, the Makana team was able to quickly generate hundreds of test cases, run evaluations, and pinpoint where their agent needed improvements.

Makana puts together a batch test plan to organize their testing strategy and ensure coverage across their Opportunity Creation workflow. They address the scenario of an employee quickly logging a new deal and assigning resources.

Batch Test Plan		
Scenario	Opportunity Creation and Resource Assignment	
Agent	Makana Opportunity Agent	
Test Type *Consider the type of tests you want to generate.*	☑ Topics and Actions *Use when you want to verify that your agent correctly routes conversations using predefined topics and actions, such as opening cases, escalating issues, or updating records.* ☐ Knowledge-Based Question and Answer *Use to verify the agent retrieves accurate, conversational answers from knowledge bases, like uploaded PDFs or knowledge articles.*	
Goal *What are you trying to validate? What specific knowledge do you want the agent to retrieve and how should it be presented?*	Ensure the agent reliably executes the full Opportunity Creation topic workflow: gathering necessary inputs, running the flow-based Create Opportunity action, running the prompt-based Identify Recommended Sales Team action, and running the standard Create a To-Do action.	
Agent Setup *Which agent are you testing and what actions, topics or resources is it connected to?*	Topics: • Opportunity Creation Actions: • Create Opportunity • Identify Recommended Sales Team • Create a To-Do	Data Library Resources:
Test Description *Describe the purpose of the test and scope like what the utterances should trigger or cover. Include key terms, tone types, and any constraints.*	A sales rep needs to quickly log a new deal for an existing account and ensure internal sales team resources are automatically identified and assigned follow-up tasks. Test utterances should include variations in project description, close dates, and tone.	
Context Variables *Identify and include the variables that*	currentRecordId: To ensure the agent correctly links the new opportunity to the Account the user is currently viewing.	

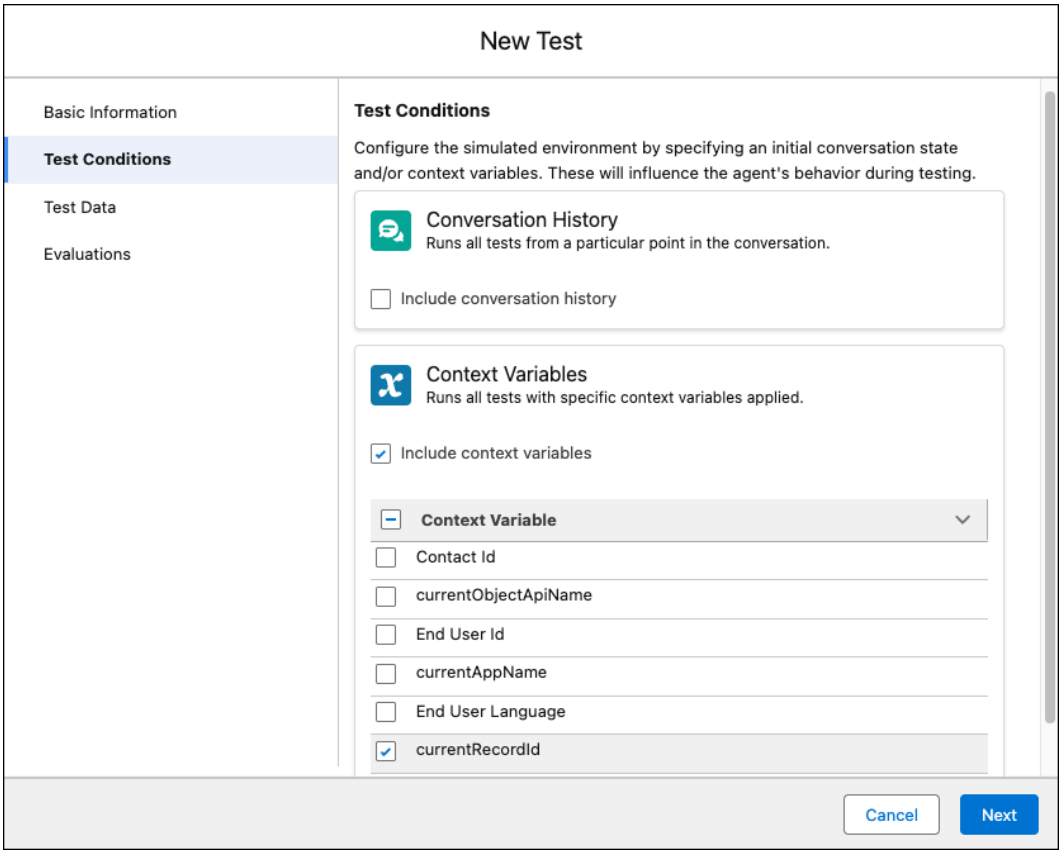
Batch Test Plan	
complement your scenario.	
Response Quality Evaluations *Choose what to measure based on the scenario. Describe the purpose of each evaluation.*	
☑ **Completeness**	Selected to verify that the agent successfully gathers all required opportunity fields (e.g., project name, close date, description, next step) via the Create Opportunity action inputs and correctly generates recommended team output.
☑ **Coherence**	Selected to ensure the final confirmation message summarizing the creation and task assignment is conversational, clear, and easy for the sales rep to understand, rather than delivering raw data.
☐ **Conciseness**	
☐ **Latency**	

Based on this plan, Makana follows these steps to create their batch tests:

1. In the Agentforce Testing Center, the Makana team selects **New Test**.
2. They select their agent, **Makana Customer Support Agent**, and name the test `Opportunity Creation and Resource Assignment`.

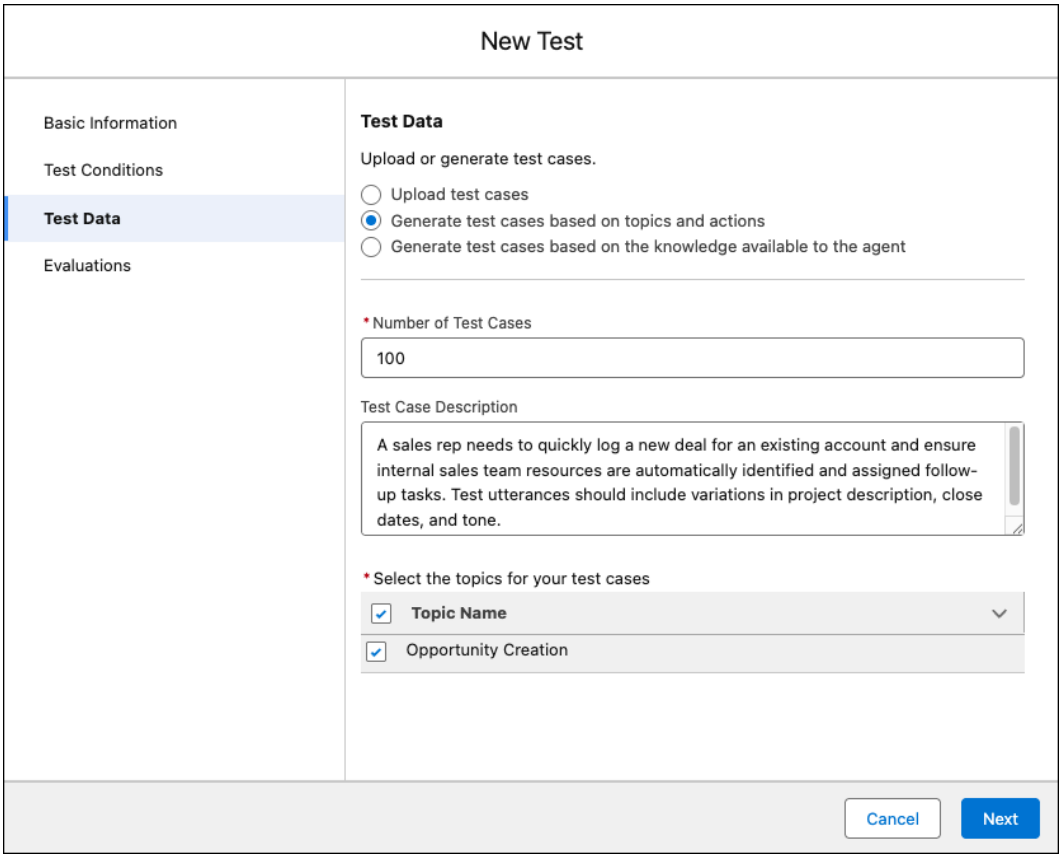
When setting up the test, Makana can include context variables to more accurately simulate real-world support scenarios. They opt not to include the conversation history.

3. Makana selects **Include context variables** and adds the **currentRecordId**. This links the new opportunity to the Account context the user is currently on.



Makana wants to validate their agent's conversation flow, topic recognition, and CRM-integrated actions.

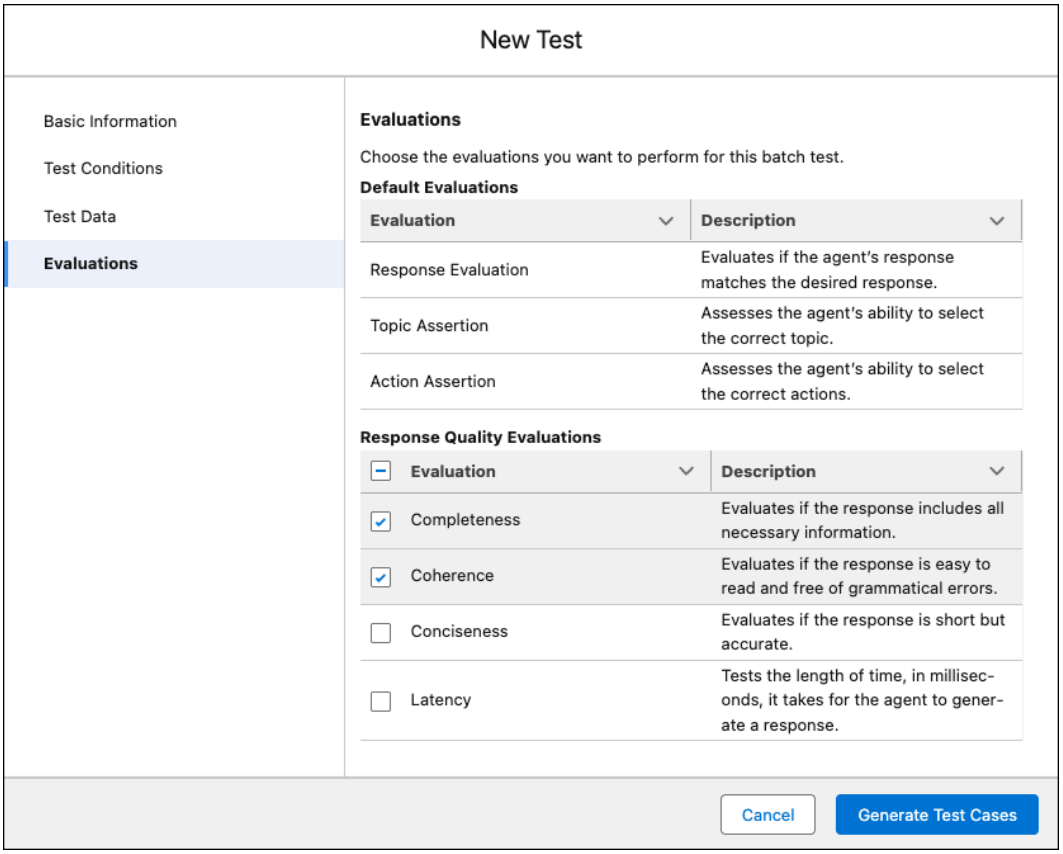
4. Makana opts to generate test cases based on the key topics and actions tied to their business objectives in their Opportunity Creation and Resource Assignment testing scenario. So, they select **Generate test cases based on topics and actions**.



5. They want a broad selection of tests, so they ask the system to create the maximum 100 tests.
6. To guide the AI in generating appropriate test cases, they enter their test description: `"A sales rep needs to quickly log a new deal for an existing account and ensure internal sales team resources are automatically identified and assigned follow-up tasks. Test utterances should include variations in project description, close dates, and tone."`
7. They include the relevant **Opportunity Creation** topic to ensure tests are focused on this specific, critical workflow.

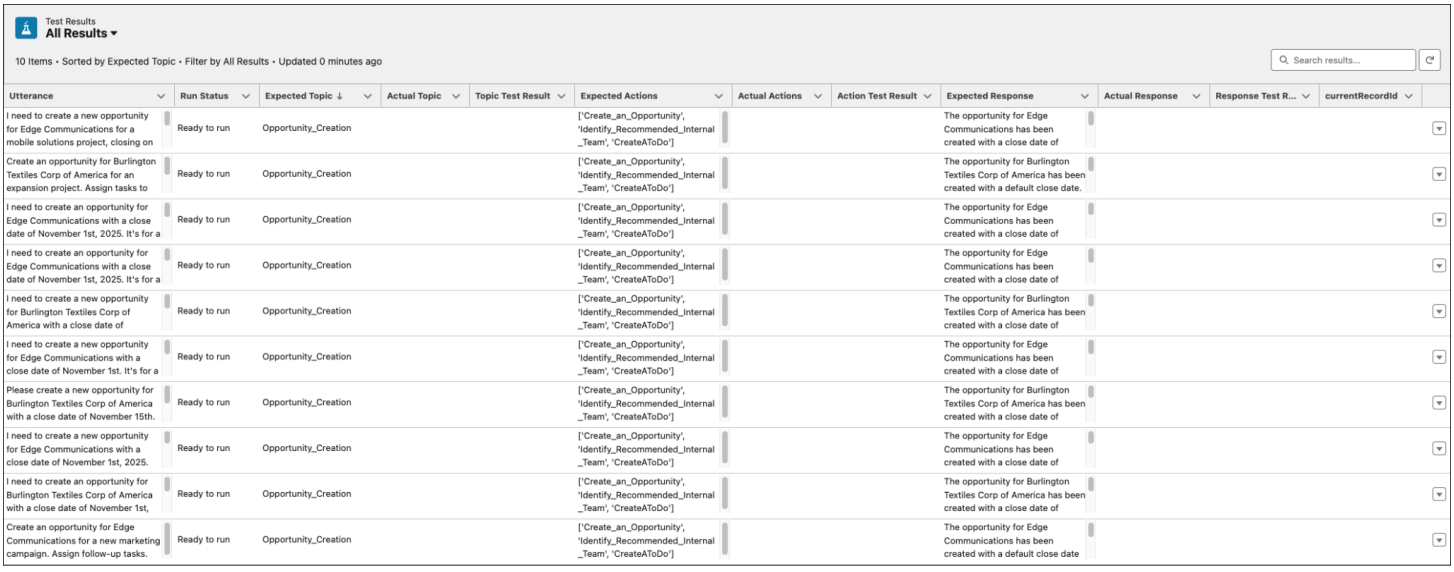
In the Opportunity Creation and Resource Assignment scenario, Makana's testing goal is to ensure the correct topics and actions are selected, and that the structured response, including recommended team roles and reasons, is communicated to the employee in a conversational, and brand-aligned manner.

8. To evaluate their responses against these goals, for the Response Quality Evaluations the Makana admin selects **Completeness** and **Coherence** for the Response Quality Evaluations.



9. Finally, they check over their test selections and click **Generate Test Cases.**

When Agentforce finishes generating the tests, they'll appear in the Testing Center.



When the status says Ready to Run, Makana clicks **Run Test Suite**. Agentforce begins running their chosen evaluations on the test cases.

Use Test Results to Improve Your Agent

Once your tests have finished running, you'll see overall metrics for the full test suite along with individual scores for each evaluation. Use these insights to identify failed or low-scoring utterances, then manually retest them in Agentforce Builder. Refine the instructions within your topics and actions as needed, and continue iterating until you reach your target level of accuracy. Pay close attention to recurring failures, drilling into these results can help uncover root causes such as unclear guidance, missing knowledge, or configuration issues. To interpret your results, see how each evaluation helps to measure agent performance.

Default evaluations are given a score of 1, pass, or 0, fail.

Evaluation	Testing Significance
Response Evaluation	Scores of 5 indicate that the agent generally succeeds in achieving its goal. A score of 5 reflects a precise, complete, and brand-aligned response with no irrelevant content. Scores of 3 to 4 show decreasing levels of clarity and completeness. There may be minor omissions or partial understanding which introduces some ambiguity while the core intent is still addressed. Scores of 1 to 2 suggest the agent struggled to meet the goal. The response may be unclear, missing key elements, or include irrelevant information. It might also ask the user for information that should have been retrieved from the CRM. A score of 0 represents a complete failure to resolve the user's query. The response is generic and doesn't address the user's intent. **Tip:** Check your agent configuration for issues related to topic selection, instructions, or actions. Consider whether there are knowledge gaps, like outdated articles, or other issues preventing the agent from responding appropriately.
Topic Assertion	A score of 1 indicates that the agent correctly identified the appropriate topic to address an utterance. A score of 0 indicates that the agent selected an unexpected topic to address an utterance. **Tip:** Manually retest any failed utterance in Agentforce Builder and review the agent's reasoning in the plan canvas. Refine the instructions for both the expected action and the topic itself to

	clearly guide the agent toward the correct choice and restrict the use of the incorrect action.
Action Assertion	A score of 1 indicates that the agent correctly identified all of the appropriate actions within a topic to address an utterance. A score of 0 indicates that the agent either chose the wrong actions or failed to select all necessary actions within a topic to address an utterance. **Tip:** Manually retest any failed utterance in Agentforce Builder and review the agent's reasoning in the plan canvas. Refine the instructions for both the expected action and the topic itself to clearly guide the agent toward the correct choice and restrict the use of the incorrect action.

Response quality evaluations are primarily scored between 0 and 5, with 3 or higher signifying a pass; however, Instruction Adherence is uniquely scored as High, Low, or Uncertain. The reasoning for each score is provided by the LLM judge via an infobubble.

> ***What is an LLM-as-judge?***
> LLM-as-Judge is when one large language model (LLM) evaluates the outputs of another LLM, serving as a scalable, automated, and objective evaluation tool for tasks like scoring summaries or ranking responses. A judge LLM receives a prompt that includes the task and evaluation criteria like, factual accuracy, relevance, coherence, and faithfulness to source. With these resources and guidelines, the LLM judge determines the expected response and compares it to the agent response then generates scores, rankings, or textual feedback. We've carefully designed our LLM-as-Judge prompts to give you the most accurate and useful test results.

Evaluation	Testing Significance
Completeness	A score of 5 indicates a fully complete and accurate answer with no important omissions. Scores of 3 to 4 reflect decreasing levels of completeness, with minor to moderate gaps that may slightly affect understanding. Scores of 1 to 2 indicate that the generated answer is significantly incomplete, missing several or most important pieces of information, which may cause confusion or a misleading result.

	A score of 0 represents a failure, signifying the answer missed all the important pieces of information and is highly confusing or misleading.
Coherence	Scores from 3 to 5 indicate that the agent correctly transforms underlying information into conversational language with the appropriate sentence and grammatical structures, ensuring that the dialogue flows smoothly and is easily understood by the user. Scores from 0 to 2 indicate the response is grammatically incorrect and unclear, or information has been taken from Salesforce objects and delivered as raw data like JSON structures or direct field content.
Conciseness	Scores from 3 to 5 indicate that the agent's response is short but accurate, successfully capturing the essence of the desired content. Scores from 0 to 2 mean the generated response is lengthy, repetitive, includes unimportant points, or contains irrelevant content.
Latency	A high or unusual latency in a test utterance suggests a problem with either the utterance itself or some underlying infrastructure. If agent adjustments don't resolve the issue, contact Salesforce support to check for infrastructure issues.
Instruction Adherence	**High:** The agent interprets and fully follows the topic instructions, both addressing key points and providing any required information. **Low:** The agent doesn’t interpret or follow the topic instructions accurately. It fails to follow at least one instruction, leading to an incorrect response. This flags a need to refine instructions and set clearer constraints. **Uncertain:** Instruction adherence can't be determined due to an ambiguous response or incomplete response, or conflicting interpretations of the topic instructions.

While default evaluations are scored by comparing expected and actual results, quality evaluations are assessed by the judge-LLM based on fixed criteria. If a response receives a low score, review it carefully to see whether it truly falls short of your expectations. For example, the judge-LLM may assign a low conciseness score, but you might feel that additional context better serves your customers or aligns with your brand voice. Otherwise, you can add extra instructions or guardrails to better tailor the agent to your goals. Always consider your specific goals and use case when

interpreting scores. Quality evaluations provide guidance, but they are not absolute measures of your agent's success or failure.

Let's revisit Makana's Case Status Inquiry batch test to see how they used evaluation results to improve their agent. After running a batch test suite, the results revealed a recurring pattern of failures tied to the Coherence quality evaluation. While the Topic Assertion and Action Assertion scored high, indicating the correct multi-action workflow was triggered: Create Opportunity, Identify Recommended Sales Team, and Create a To-Do, the final response failed the quality check because the agent provided raw data instead of a readable summary.

Utterance	Expected Topic	Expected Action	Actual Topic	Actual Action	Result
"Log the new potential sale for Beta Systems. Set the close date for next month."	Opportunity Creation	Create Opportunity	Opportunity Creation	Create Opportunity	Fail (0)

The Coherence score of 1 indicates that the agent delivered the output information as raw data like JSON structures or direct field content instead of conversational language. For example, the agent may have output the raw opportunity record ID (006A000...) and the JSON structure defining the recommended roles, making the result unusable for the sales representative.

Makana follows the recommended process for using failed test results to troubleshoot the agent:

1. *Manual Retest in Agentforce Builder*
   Makana copies the failed utterance, `Log the new potential sale for Beta Systems. Set the close date for next month.` and enters it into the Conversation Preview panel in Agentforce Builder.
2. *Review and Analysis*
   In the plan canvas, the team confirms that the agent correctly selected the Opportunity Creation topic and executed the three required actions in sequence: Create Opportunity, Identify Recommended Sales Team, and Create a To-Do. By reviewing the agent's final generated output in the Conversation Preview panel, they observe that the agent successfully created the opportunity record in Salesforce but failed to summarize the outcome conversationally. Instead, it returned the raw output variables, Id, Name, and NextStep fields stored in the opportunityCreated record variable, without proper formatting.
3. *Refine Action and Topic Instructions*
   Since the issue is related to transforming the underlying Salesforce data into conversational language, Makana focuses on refining the action instructions. They adjust the instructions for the Create Opportunity action to guide the agent on how to present the data: `Summarize the opportunity details created (ID, Name, and Next Step), and confirm that the recommended sales team members were identified`

`and follow-up tasks were created. The summary must be clear and conversational.` This refinement ensures the agent uses the new opportunity record fields to generate a coherent, user-friendly response, rather than delivering raw data.

4. *Verification and Re-testing*
   The team restarts the conversation in the Agentforce Builder and enters the failed utterance again. They confirm that the agent now correctly performs the multi-step workflow and provides a concise confirmation message that smoothly integrates the results of all three actions, achieving a high Coherence score.
5. *Scaling the Fix*
   To ensure the fix reliably addresses similar transactional inquiries and maintains conversational quality, Makana runs another batch test using cases designed to validate the successful execution and subsequent coherent communication of the full Opportunity Creation workflow. They track the Coherence score over time to prevent agent drift.

Iteration and Refinement

Agent testing isn't a one-and-done project. New features roll out, new data is added, and LLMs are updated, all of which can contribute to your agent's behavior changing over time. Regular batch testing and refinement is critical to make sure the agent is still performing well and meeting your business objectives.

Batch test results offer more than just a simple pass/fail; they provide valuable insights for proactively refining your agents. By leveraging various metrics and strategic actions, you can transform agent maintenance from a reactive, bug-fixing process into a proactive, data-driven cycle of continuous improvement.

The Testing Center provides a crucial capability for maintaining the quality and performance of your agents: the ability to save and track batch test results over time. This historical data is invaluable for identifying agent drift, a phenomenon where an agent's performance gradually degrades due to various factors.

By regularly executing the same comprehensive test suite against your agents, you establish a baseline for their expected behavior and performance. This consistent testing allows you to spot even subtle behavioral changes that might arise from updates to underlying large language models (LLMs) or modifications to data access. For instance, a new LLM version, while generally improved, might introduce regressions in specific, niche scenarios that your existing agent handles well. Similarly, changes to data sources or retrieval methods could inadvertently impact an agent's ability to process information accurately.

To accurately measure the impact of any major changes you introduce to your agents–whether it's an architectural overhaul, a significant data update, or a new LLM integration–it is imperative to always run a batch test before and after the change. This pre- and post-comparison provides a clear, quantitative assessment of the modification's effect on performance. If the post-change test results reveal a performance drop, you gain immediate visibility into the issue. This rapid detection

is critical, enabling you to quickly roll back the problematic change or refine it until performance is restored. This proactive approach prevents regressions from reaching production and negatively impacting user experience, ultimately safeguarding the reliability and effectiveness of your agents.

While batch testing is powerful, it can't capture everything. It's important to integrate human feedback into your process. For example, a batch test might confirm factual correctness, but a human can judge tone and empathy in an utterance. Teams can then flag specific interactions for review.

To further enhance your testing and refinement process, versioning is essential for isolating changes and finding the source of issues. By creating a version of your agent, you can make changes to it and test them without having to take an agent out of production. You can create up to 20 versions of your agent and use them to test different parts of a major change. By treating every change as a distinct, traceable unit, you can rewind your project's history, compare different versions, and pinpoint exactly when a problem was introduced. This turns the chaotic process of debugging into a structured, forensic investigation.

To make versioning a powerful debugging tool, you need to follow a few key practices. Each version should represent a single, logical change. Instead of performing multiple bug fixes and refactors all at once, break it down into small, focused changes. This makes it easier to isolate the exact issue that caused a bug. Additionally, make sure you capture what each change was and why you made it. It's a breadcrumb trail for your future self and other collaborators.

Learn More: *For more information about debugging agents, see Troubleshooting Agents.*

Deploy Your Agent

Following Makana Medical Devices' successful ideate, build, and test phases, the deploy stage outlines the step-by-step process for deploying an Agentforce Employee agent to Lightning Experience, the Salesforce Mobile app, and Slack.

In the deploy stage, you prepare your AI agent for your employees. This process involves launching your deployment strategy, setting up the agent's connections, and deploying your agent. A deployment strategy helps you maximize your AI agent's success, minimize disruption, anticipate challenges, manage stakeholder expectations, and gather critical feedback for continuous

improvement. The components of an effective deployment strategy include a rollout plan, which often begins with a pilot program. After refining the agent's performance, you can gradually roll it out to all of your employees.

AI projects also require robust change management strategies to reduce risk and ensure successful adoption. Factors that impact adoption of AI agents include AI maturity, organizational readiness, workflow integration for AI tools, ethical AI governance, training plans, effective communication, executive sponsorship, a culture of AI experimentation and transparency, and feedback channels. For more information about agent deployment strategies, see Agentforce Deployment: Quick Look.

Starting here? *Make sure you have the right permissions for the job.*

Before You Begin

- Download and log into the Salesforce Mobile App
- Sign up for Slack free tier

Learn About Connections

Channels are the platforms, apps, and interfaces that you can deploy an agent to, such as the Agentforce panel in Lightning Experience, enhanced messaging channels, Slack, and email.

Connections help you scale agent development and reduce repetitive setup by letting you build an agent once and easily add it to multiple channels. A connection gives the agent context about how to reason and respond appropriately for one or more channels. Each connection contains adaptive response formats that tell an agent how to structure responses and deliver multimedia content, such as images, buttons, links, and videos, on a variety of channels. You can also find the Omni-Channel flows that route conversations to and from an agent in your connection settings.

For the Makana Opportunity agent, you'll use the Messaging connection to connect the agent to Slack. The Messaging connection is associated with enhanced Messaging channels and external apps, including Slack, and is automatically included with all Agentforce Employee agents. When an employee sends a message to the Makana Opportunity agent in Slack, the agent starts using the Messaging connection and adaptive response formats to reason and respond.

Learn More: *For more information about agent deployment concepts, see Deploy Your Agent to Channels.*

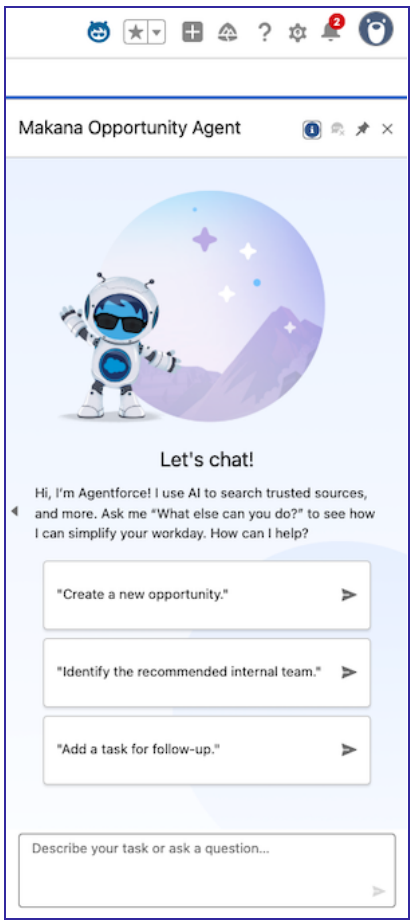
Deploy the Agent to Lightning Experience and Mobile

When you activate your agent, it becomes available in the Agentforce panel in Lightning Experience and in the Salesforce mobile app for all employees with access. If an employee has access to more than one agent in the Agentforce panel, they can select a specific agent to interact

with from the menu. To make sure that the Makana Opportunity agent is ready for your employees, test it in Lightning Experience and the mobile app.

Test the Agent in Lightning Experience

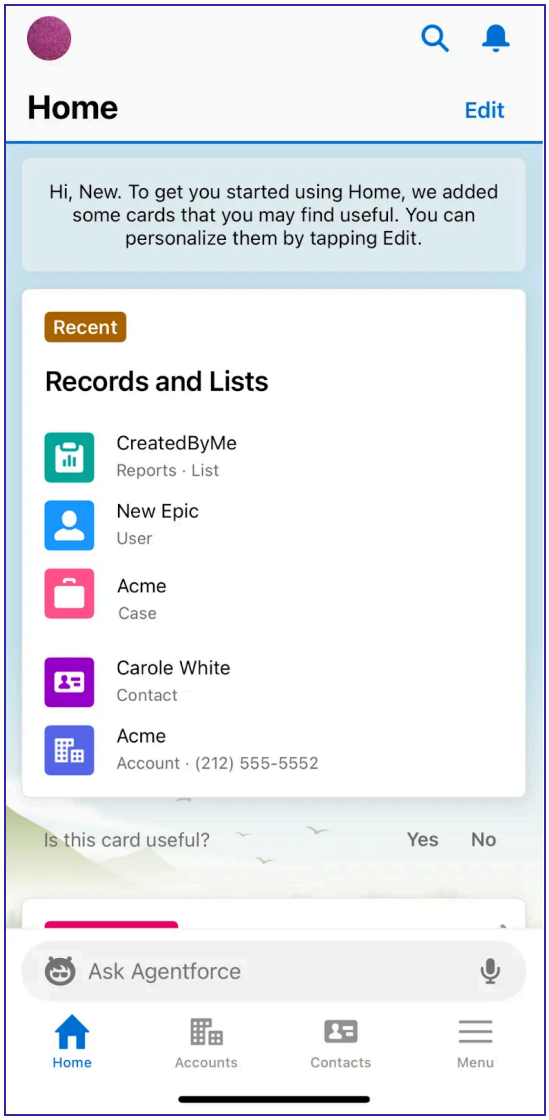
1. Open Salesforce on a desktop.
2. To open the Agentforce panel, click the Agentforce icon in the header.



3. In the text box, enter `Create a new opportunity for Acme Corp for the Pro BP-200.`

Test the Agent in Mobile

1. Open the Salesforce mobile app.



2. In the Ask Agentforce box, enter `Update the Acme Corp Pro BP-200 opportunity with a close date 30 days from today.`
3. Tap the microphone icon. If you haven't already, give the app access to speech recognition and microphone and opt in to using Agentforce. Then tell Agentforce, `Also update the opportunity with a potential $50,000 deal.`

Using an iOS device? *Here are three more ways that you can access Agentforce.*

- *Add the Open Agentforce and Talk to Agentforce shortcuts in the Shortcut App.*
- *Add the Agentforce widget to the home screen.*
- *Say "Hey Siri, Ask Agentforce."*

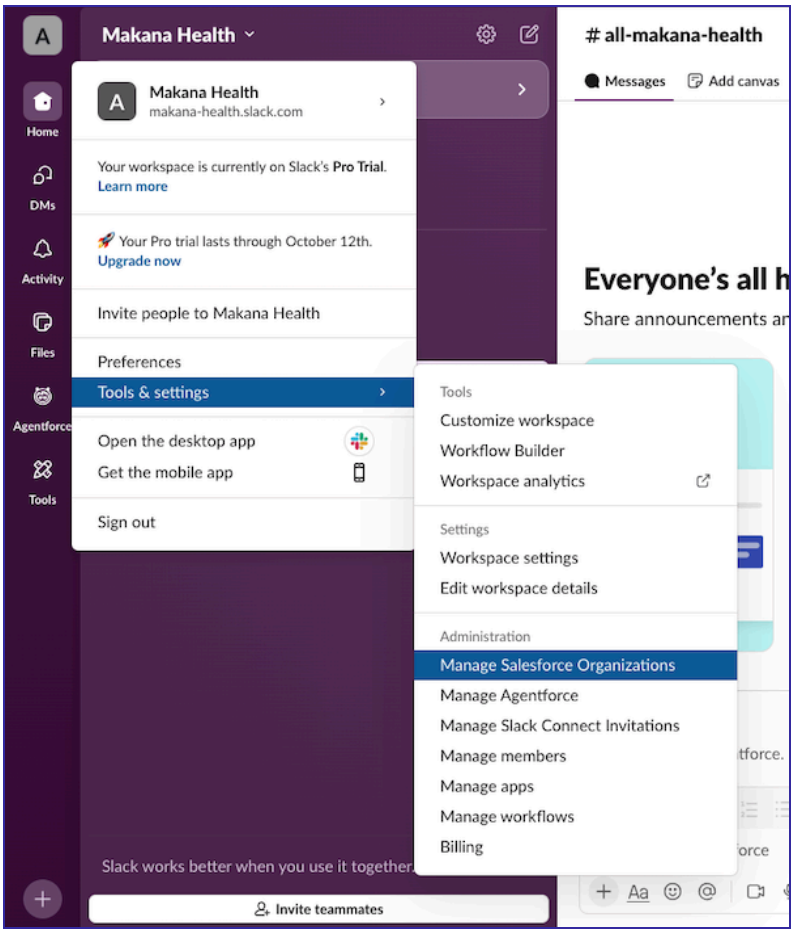
Learn More: *For more information about Agentforce in mobile, see Agentforce on the Salesforce Mobile App: Quick Look.*

Connect Salesforce and Slack

Before setting up Agentforce in Slack, connect your Slack workspace to your Salesforce org. Request, approve, and activate a connection. Then, install the Slack connected app in Salesforce. You can automatically map your employees' Slack and Salesforce accounts when you request a connection, or you can manually map them later.

Request and Approve a Connection

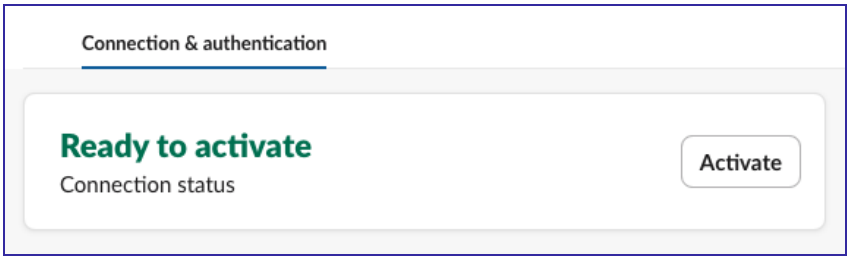
1. From the Home tab in Slack, click the name of your Slack workspace.
2. From the dropdown, hover over **Tools & settings** and then select **Manage Salesforce Organizations**.



3. Click **Connect Salesforce Org** and then enter your Salesforce org URL.
4. For the Account mapping field, select **Email** or **SAML NameID**. Or, if you want to manually map your employees' accounts later, turn off automatic account mapping.
5. Click **Request Connection** and then click **Okay**. A request is sent to the Salesforce org.
6. In Salesforce Setup, in the Quick Find box, enter `Slack` and then select **Manage Slack Connection**.
7. Select **Agree to terms, including the Slack user terms** and **I agree to the following Salesforce Channels disclosure**. Then, click **Approve**.

Activate the Connection in Slack

1. From the Home tab in Slack, click the name of your Slack Workspace.
2. From the dropdown, hover over **Tools & settings** and then select **Manage Salesforce Organizations**.
3. In the Connect Status column for your Salesforce org, click **View details**.
4. On the Connection status tile, click **Activate**. Then, on the Activate this connection window, click **Activate**.



Connect Your Employees' Salesforce and Slack Accounts

If you didn't automatically map your employees' accounts when you requested a connection, map their accounts now. Make sure that you add your user as a member.

1. From the Home tab in Slack, click the name of your Slack Workspace.
2. If you have a Pro or Business+ plan:
    a. From the dropdown, hover over **Tools & settings** and then select **Workspace Settings**.
    b. From the left sidebar, select **Salesforce organizations**.
3. If you have an Enterprise plan:
    a. From the dropdown, hover over **Tools & settings** and then select **Organization Settings**.
    b. On the left-hand side, click **Salesforce** and then select **Salesforce organizations**.
4. Click the **Users** tab.
5. Select **Add individually and search for the member you'd like to add**. To add members in bulk, click **Add by CSV** and follow the steps.

Deploy the Agent to Slack

Now that your Slack workspace and Salesforce org are connected, you can deploy the Makana Opportunity Agent to Slack. Add the Slack connected app to your agent's connections in Agentforce Builder. Then install your agent in Slack and customize the agent's profile with suggested prompts. When you're ready, test the agent.

Add the Slack Connected App to the Agent

1. From Setup, in the Quick Find box, enter `Agents`, and then select **Agentforce Agents**.
2. If your agent is deactivated, activate it.
3. Click **Makana Opportunity Agent** to open its Details page. Then click **Open in Builder**.
4. In Agentforce Builder, click the **Connections** tab. If you haven't already, turn on the new Connections panel experience.

5. On the Connections panel, select **Messaging**.
6. In the External Apps section, click **Add External App**.
    a. Select the **API** connection type.
    b. For the integration name, enter `MakanaOpportunityAgent`.
    c. For the connected app, select **Slack**.

Troubleshooting: *If Slack isn't an option in the Connected App dropdown, install the Slack package and try again.*

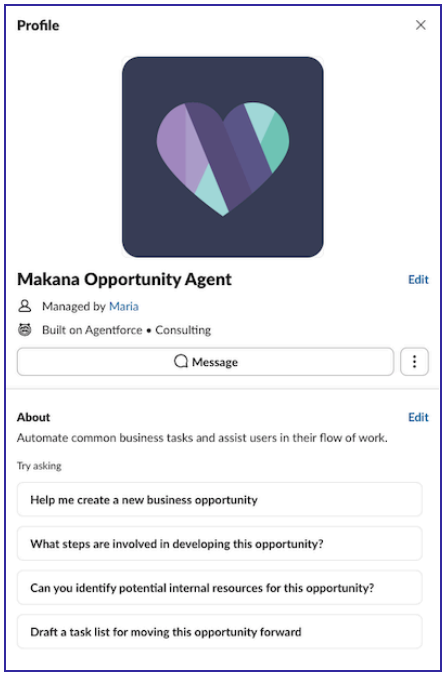
1. *To install the package, append your Salesforce URL with* `packaging/installPackage.apexp?p0=04tKX000000cHXG`. *For example,* `https://orgfarm-abcdefg123-dev-ed.develop.my.salesforce.com/packaging/installPackage.apexp?p0=04tKX000000cHXG`.
2. *Then, select* ***Install for all Users*** *and* ***I acknowledge that I'm installing a Non-Salesforce Application that is not authorized for distribution as part of Salesforce's AppExchange Partner Program****.*
3. *Click* ***Upgrade****.*
4. *Refresh Agentforce Builder and try adding Slack to your agent again.*

Install the Agent in Slack

1. From the Home tab in Slack, click the name of your Slack Workspace. Then, navigate to the Agentforce section of the Slack Admin Dashboard.
    a. If you have a Pro or Business+ Plan, hover over **Tools & settings** and then click **Manage Agentforce**.
    b. If you have an Enterprise plan, hover over **Tools & settings** and then click **Organization settings**. On the left-hand side, click **Salesforce** and then select **Agentforce**.
2. Next to the Makana Opportunity agent, click **Review**.
3. Review the permissions the agent will have in Slack, then click **Allow**.
4. If you have an Enterprise plan, check the box next to the workspace that you want the agent to be available in. Then click **Add to Workspaces**.

Customize the Agent Profile

1. From the sidebar, click .
2. Click the **Managed by you** tab.
3. Next to the Makana Opportunity agent, click the three dots icon and then select **View Agent Profile**.



4. To adjust suggested prompts, click **Edit** in the About section. Add these four prompts:
   - `Help me create a new business opportunity`
   - `What steps are involved in developing this opportunity?`
   - `Can you identify potential internal resources for this opportunity?`
   - `Draft a task list for moving this opportunity forward`

Test the Agent in Slack

1. Refresh your Slack Workspace.
2. From the sidebar, click . Then select the Makana Opportunity agent. If you don't see the Agentforce icon or see the Makana Opportunity agent, make sure that you added your user as a member in Slack and try again.
3. To start a conversation with the agent, choose one of the prompts you created or send this message: `Tell me about the Acme Corp Pro BP-200 opportunity`
4. To add the agent to a Slack channel, open a channel and then click the channel name.
5. In the window, click the **Agents** tab. Next to the Makana Opportunity agent, click **+**.
6. In the channel, enter and then send this message: `@Makana Opportunity Agent Update the Acme Corp Pro BP-200 opportunity close date to 10 days from today`. The agent responds in the thread and sends you a direct message.

Monitor Your Agent

Makana Medical Devices has successfully deployed their customer service agent to handle inquiries about their CGM-3000 continuous glucose monitor. Now they need to ensure the agent performs well and delivers the business value they expect.

This guide covers how to test, measure, and effectively optimize your AI agent's performance in a real-life environment after it's been launched.

Starting here? *Make sure you have the right permissions for the job.*

Before We Begin

This chapter is meant to provide guidance around building a continuous monitoring strategy and using Agentforce tools to monitor and improve your agents. Since we just launched the Makana Support agent, there won't be enough data to walk through a full monitoring strategy with live data.

Additionally, although we have a legacy Agentforce Analytics, we're upgrading it to a new Agentforce Analytics and Optimization solution. These updates are in Beta and not yet available for Developer orgs. Once you roll out your agents in production, use this section as a resource to keep key stakeholders informed and handle top issues.

How it Works

Analytics and Optimization go beyond simply observing performance. They help you find areas to improve. You can refine agent behavior to meet goals and improve user satisfaction. Additionally, you can also iterate on agent prompts and knowledge bases.

Agentforce Analytics provides a broad, comprehensive view of agent sessions with meaningful insights into agent functions such as interaction latency, usage patterns, and performance metrics, like overall quality scores and success rates. Analytics answers foundational questions like "Is my agent working as expected?" and "What can I do better?"

Agentforce Optimization complements Analytics by providing moment-specific data for granular inspection. Moments refer to sub-sessions that capture interactions addressing specific user intent within broader sessions. An LLM identifies similar intents across multiple sessions and divides them

into moments. The division into moments makes inspection more granular and helps you identify instances of agents providing inaccurate, misleading, or irrelevant information, or failing to execute actions as configured.

Set Up Audit and Feedback

Turn on Audit and Feedback to store details about interactions with LLMs for analysis and reporting. Audit data captures prompts and generated AI responses (including PII-masked versions and toxicity scores). Feedback captures explicit user feedback about generated responses, such as "thumbs up/down" ratings and reason text. Use insights derived from this information to monitor, test, troubleshoot, and improve your agents in Data Cloud. To learn more, see Generative AI Audit and Feedback Data.

By turning on these features, you consent to store your Salesforce organization's generative AI activity log and feedback data in Data Cloud. Consent also assumes you've reviewed any potential cost implications and assume responsibility for any associated costs.

1. From Setup, in the Quick Find box, enter `Einstein Audit`, and then select **Einstein Audit, Analytics, and Monitoring Setup**.
2. Turn on **Audit and Feedback**.
3. If you have multiple data spaces in your org, select a target data space in which to store the audit and feedback data.
4. Verify that the pre-built reports and dashboards are installed and available. It can take a few minutes for the installation to complete.
    a. Click **App Launcher**, and in the search field, enter `Data Cloud`.
    b. To see audit and feedback dashboards, from Data Cloud, click **Dashboards**. Under FOLDERS, click **All Folders**. In the Search all folders search box field, enter `Einstein Gen AI Audit & Feedback.`
    c. To see audit and feedback reports, from Data Cloud, click **Reports**. Under FOLDERS, click **All Folders**. In the Search all folders search box field, enter `Einstein Gen AI Audit & Feedback.`

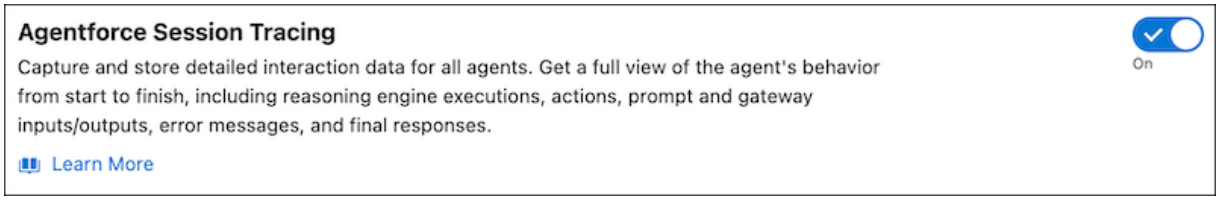
After Einstein Generative AI data collection is enabled, it can take a few minutes for the reports and dashboards to become available. If for any reason the reports and dashboards are not installed after enabling data collection, the system will continue to retry the installation.

Set Up and Configure Agentforce Analytics

Agentforce Analytics is designed for comprehensive session analysis, capturing every turn and event across your agent sessions using the unified Session Tracing Data Model (STDM). Agentforce Analytics uses Tableau Next for its dashboards visualization. To view them in Agentforce Analytics, you must manually install the libraries in addition to the setup steps.

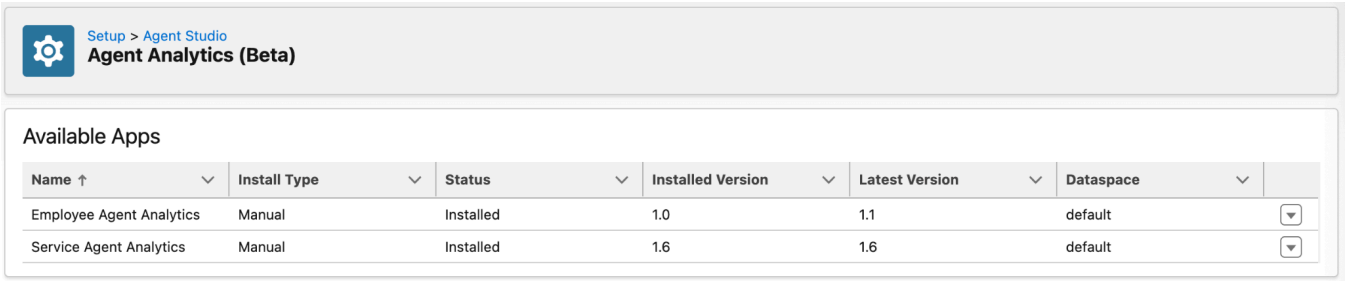
To set up Agent Analytics, turn on Session Tracing to collect and store session data. Session Tracing also provisions the required Analytics model entities to perform queries.

1. Verify that you have the latest version of the Salesforce Standard Data Model (version 1.107 or higher) in your org (sandbox or production). From Setup, go to **Apps** -> **Packaging** -> **Installed Packages** -> **Salesforce Standard Data Model**. Get the required version, and follow the instructions.
2. From Setup, in the Quick Find box, enter `Einstein Audit, Analytics, and Monitoring Setup`, then select **Einstein Audit, Analytics, and Monitoring Setup**.
3. Scroll down to **Agentforce Session Tracing and Data Model**.
4. Toggle on **Agentforce Session Tracing and Data Model** to start collecting data and to provision the Data Model Objects. Note that turning on data collection increases your org's credit consumption rate.



5. From Setup, go to **Agent Analytics (Beta)**.

6. Install the dashboard libraries for your available agents. Dashboards are created from Tableau Next.



7. Refresh the page to see the status changed to **Installed**.

Observe and Optimize with Agentforce Analytics and Optimization

Once configured, Agentforce Analytics and Optimization provides tools for continuous monitoring and improvement of AI agents. Optimization extends session tracing by focusing on session moments, which are sub-sessions representing interactions with a specific user intent. An advanced LLM identifies clusters, and tags these moments, facilitating queries and insights.

To add Agentforce Optimization to your observability tools, toggle on **Agentforce Optimization** from setup in addition to **Agentforce Session Tracing and Data Model** and assign users the relevant permission sets.

1. From Setup, go to **Einstein Audit, Analytics, and Monitoring Setup**.

2. Toggle on **Agentforce Session Tracing and Data Model and Agentforce Optimization** (Optimization relies on session tracing as well).

The foundation data model for Analytics provides insights and metrics that Makana's team can use to reach their goals.

Analytics Data Model and Semantic Layer

The foundation for the Analytics data model is the unified Session Tracing Data Model (STDM), which captures every logged event within a session, down to a single conversation turn.
The Analytics Semantic Layer links Session Tracing entities with Optimization data entities, enabling complex queries. You can access this in Data Cloud under the semantic model tab by opening Agentforce Analytics Foundations.

Analytics Queries and Calculated Fields

The Semantic Layer provides calculated fields for querying session and agent performance data. These measures provide quantitative insights:

- Quality_Score_clc & Average_Quality_Score_clc: Relevance score (1-5) indicating the accuracy/fulfillment rate of an agent's response to a user's request.
- Quality_Score_Reasoning_clc: Explanation for an assigned quality score (e.g., "agent didn't address the pricing question").
- Moment_Duration_clc: Duration of a Moment in seconds.
- Average_Moment_Duration_clc: Average duration of a Moment in seconds. This helps identify interactions that take longer.
- Escalation_Rate_clc: Percentage of sessions escalated to a human or different agent.
- Deflection_Rate_clc: Percentage of sessions ended by the user (not by escalation or abandonment).
- Unique_Moments_clc: Number of distinct moments recorded, clustering interactions around a user request.

You can combine multiple fields in a single query to generate granular metrics, such as the relationship between session duration and quality ratings (e.g., Session_Count, Average_Session_Duration_clc, and Quality_Score_clc) or to track performance trends over time.

For example, combining Moment_Duration_clc and Quality_Score_clc helps identify areas where users spend significant time but receive poor-quality responses.

Monitoring Strategy Methods

This sample plan shows you how to use Agentforce Analytics and Optimization for a comprehensive view of agent performance. You can also inspect specific issues in agent sessions and fix them. There are three major methods in a monitoring strategy:

Stage	When to Begin	Goal

Post-Launch Monitoring	Immediately following agent launch	Confirm agent is working as designed and metrics are being populated
Moment Monitoring	30 days after agent launch; 24 hours after a closed moment; up to 30 days for deep data insights	Identify actions for improvement
Root Cause Analysis	Minimum clustering of 100 moments	Use clusters to identify issues and dive deep into sessions

Post-Launch Monitoring

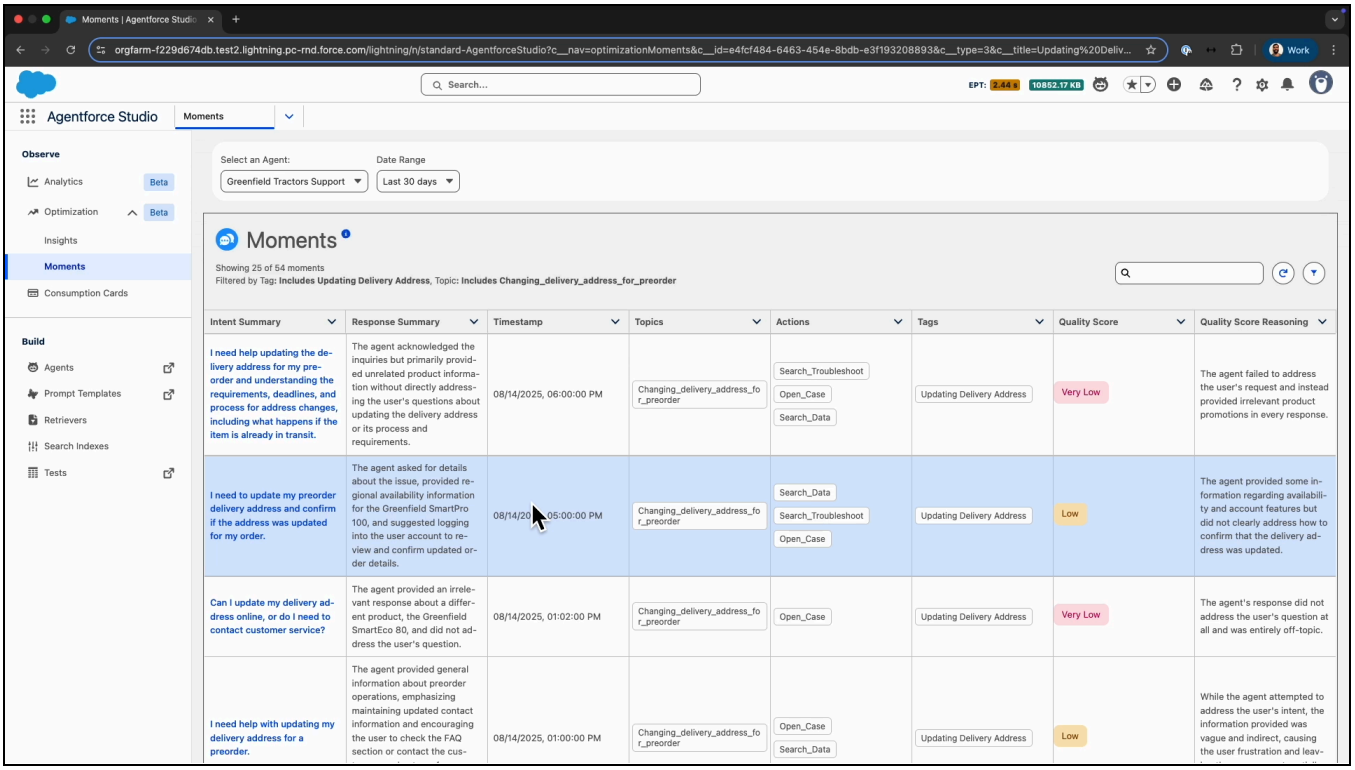
Immediately post-launch, use Agentforce Analytics to check if data is collected and ingested and view your dashboard for key metrics:

- Usage patterns (e.g., unique users, number of sessions)
- Overall performance metrics like average quality score and escalation rate
- Agent latency

You can also dive into specific sessions. Select sample sessions from key topics your sessions cover, and inspect them turn-by-turn to see if they flow as expected.

Moment Monitoring

When your system begins to cluster and tag moments within interactions, you can view the moments tab in your app and filter the moments by quality scores. If you inspect individual specific moments with low quality scores, you can uncover possible agent actions that need refinement or improvement.

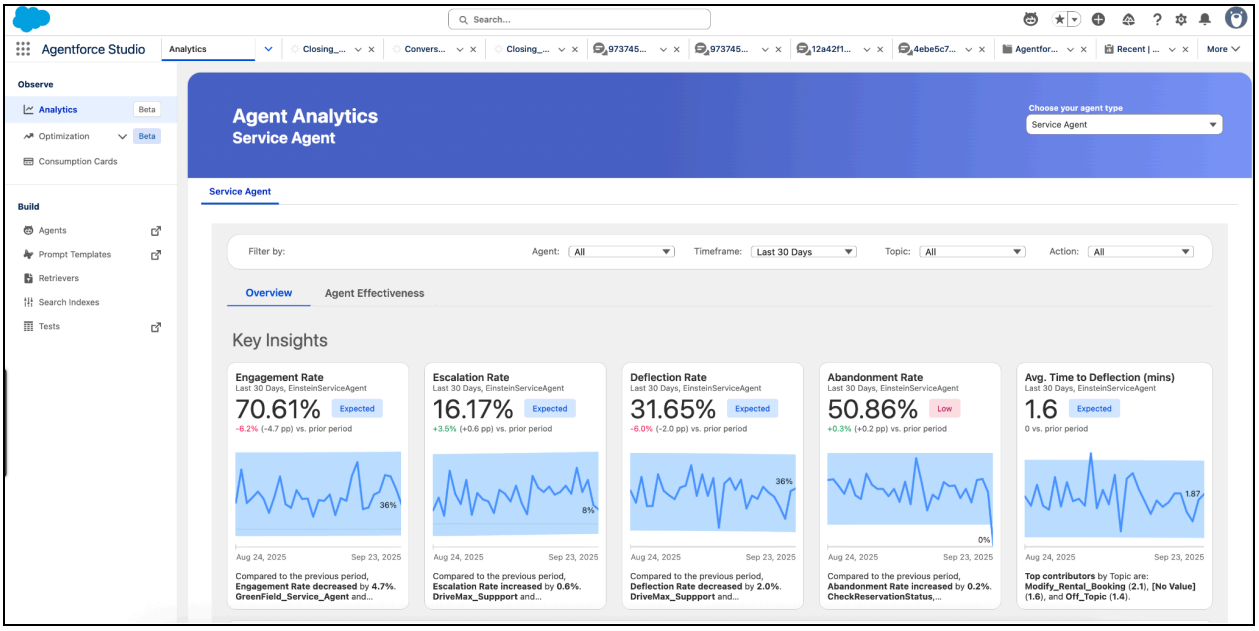


Timeline:

- Initial Data Analysis: The first runs will analyze session data from the last 30 days to provide an initial baseline. Subsequent runs will focus on sessions from the last 7 days.
- Moment Generation: The system generates moments (sub-sessions representing a specific user intent) on a frequent schedule. A pipeline run is initiated under one of two conditions:
    - Every 9 hours if there is at least one new closed session.
    - Every 3 hours if there are more than 10 new closed sessions since the last run.
    - A “closed session” is defined as a session where the last interaction occurred 24 hours ago.
- Clustering: The LLM clusters similar moments and applies tags once per week over the weekend. This process is crucial for identifying trends and widespread issues.

Root-Cause Analysis

This stage focuses on using the detailed data to perform root-cause analysis and identify specific areas for improvement. Drill Down with Agentforce Optimization and move from the "what" of Analytics to the "how" of Optimization. Use the tool to perform a granular inspection of specific moments and clusters.



Timeline

- This phase typically begins after the first weekly clustering run is complete, as this provides the necessary data for deep-dive analysis.
- For the clustering algorithm to provide meaningful results, a rule of thumb is that around 100 moments are needed. This can typically be generated from about 100 sessions, though this depends on your specific data.

Actions

1. Your main Analytics dashboard will give you the initial indication of performance gaps and flagged interactions.
1. Identify Performance Gaps: Agentforce Optimization automatically flags interactions where the agent provided inaccurate, misleading, or irrelevant information (including “hallucinations”) or failed to execute a configured action.
2. Analyze performance of individual moments according to their Quality Score:
    - To check an individual moment's quality, examine its quality score, ranked from 1(lowest) to 5(highest).
    - Then review the Quality Score Reasoning, which provides a specific explanation for the assigned score (e.g., ‘agent didn't address the pricing question’).
3. Correlate Metrics for Deeper Insights: Combine multiple calculated fields in a single query to find complex problems. For example:
    - Querying for moments with a high Moment_Duration_clc and a low Quality_Score_clc helps identify areas where users spend a lot of time but receive poor-quality responses.
    - Adding the Escalation_Rate_clc to this query can confirm that these poor interactions are leading to escalations.

Building a Monitoring Strategy

We recommend using the monitoring methods over time to build out a full monitoring strategy.

Daily

1. Monitor high-level KPIs in Agentforce Analytics. When a negative trend is spotted (e.g., rising Escalation_Rate_clc), use Agentforce Optimization to drill down and identify the specific moments or user intents causing the issue.
2. Monitor internal and external feedback about the agent. If an issue arises, perform a root cause analysis to identify the problem.

Weekly

1. Monitor moments to identify agent areas of improvement.

Monthly

1. Collaborate with Subject Matter Experts (SMEs) and AI Agent Builders to analyze the root cause identified in moments. This could be a knowledge gap, a poorly configured prompt, or an issue with the Retrieval-Augmented Generation (RAG) configuration.
2. Implement Changes: Implement the required changes, such as refining agent prompts, updating knowledge bases, or modifying agent actions and topics.
3. Test and Validate in a Sandbox Environment: before deploying improvements to production, validate all changes in your sandbox environment.
    - Manual Testing: You can interact with the agent manually via Agent Builder or a testing center to review its reasoning utterance-by-utterance, and perform manual adjustments if needed.
    - Batch Testing: For efficiency, you can run up to 100 test cases at once without deactivating the agent.
4. Redeploy & Monitor: Once the changes have been validated in the sandbox, redeploy the updated agent to production. Use Agentforce Analytics to monitor KPIs to confirm that the changes had the desired positive impact, creating a continuous cycle of learning and optimization.