Leitfaden zur Implementierung

Generativer KI-Anwendungsgenerator auf AWS

Generativer KI-Anwendungsgenerator auf AWS: Leitfaden zur Implementierung

Table of Contents

Diese Lösung erleichtert die Entwicklung, das schnelle Experimentieren und den Einsatz von Anwendungen der generativen künstlichen Intelligenz (KI)

Generative AI Application Builder auf AWS erleichtert die Entwicklung, das schnelle Experimentieren und die Bereitstellung von Anwendungen mit generativer künstlicher Intelligenz (KI), ohne dass umfangreiche KI-Erfahrung erforderlich ist. Diese AWS-Lösung beschleunigt die Entwicklung und optimiert das Experimentieren, indem sie Ihnen hilft:

- Investieren Sie Ihre geschäftsspezifischen Daten und Dokumente
- Bewerten und vergleichen Sie die Leistung großer Sprachmodelle () LLMs
- Führen Sie mehrstufige Aufgaben und Workflows mit KI-Agenten aus
- Erstellen Sie schnell erweiterbare Anwendungen und stellen Sie diese Anwendungen mit einer Architektur der Enterprise-Klasse bereit

Generative AI Application Builder auf AWS umfasst Integrationen mit:

- LLMs erhältlich bei Amazon Bedrock
- LLMs die Sie auf Amazon SageMaker AI bereitgestellt haben
- Amazon Bedrock Wissensdatenbanken für Retrieval-Augmented Generation (RAG)
- Amazon Bedrock Guardrails zur Implementierung von Schutzmaßnahmen und zur Reduzierung von Halluzinationen
- Amazon Bedrock Agents zur Erstellung agentischer Workflows, mit denen Aufgaben orchestriert und abgeschlossen werden können
- Amazon Bedrock AgentCore zur Entwicklung, Bereitstellung und Verwaltung produktionsreifer KI-Agenten mit erweiterter Runtime-Support
- Model Context Protocol (MCP) -Server für die Integration von Unternehmensdaten und Tools

Darüber hinaus ermöglicht diese Lösung mithilfe von LangChain Konnektoren Verbindungen zu einem Modell Ihrer Wahl. Diese Konnektoren sind in einer AWS Lambda Lambda-Funktion verfügbar, die zusammen mit der Lösung bereitgestellt wird. Sie können mit dem Bereitstellungsassistenten ohne Code beginnen, um generative KI-Anwendungen für die Konversationssuche, KI-generierte Chatbots, Textgenerierung und Textzusammenfassung zu erstellen.

Dieser Implementierungsleitfaden bietet einen Überblick über die Generative AI Application Builder on AWS-Lösung, ihre Referenzarchitektur und Komponenten, Überlegungen zur Planung der Bereitstellung und Konfigurationsschritte für die Bereitstellung der Lösung in der Amazon Web Services (AWS) Cloud.

Dieser Leitfaden richtet sich an Lösungsarchitekten, Geschäftsentscheider, DevOps Ingenieure, Datenwissenschaftler und Cloud-Experten, die Generative AI Application Builder auf AWS in ihrer Umgebung implementieren möchten.

Verwenden Sie diese Navigationstabelle, um schnell Antworten auf diese Fragen zu finden:

Wenn du willst.	Lesen.
Informieren Sie sich über die Kosten für den Betrieb dieser Lösung. Die geschätzten Kosten für den Betrieb dieser Lösung hängen von den bereitgestellten Komponenten und der Anzahl der Abfragen ab. Die Kosten für die Ausführung des Deployment-Dashboards mit Standardparametern und 100 aktiven Benutzern in der Region USA Ost (Nord-Virginia) für einen Monat belaufen sich auf etwa 20,12 USD pro Monat. Die Kosten für einen Text-Anwendungsfall, der ohne RAG für einen Geschäftsbenutzer bereitgestellt wird, der 100 Abfragen pro Tag mit dem LLM durchführt, belaufen sich auf etwa 12,39 USD pro Monat. Die Kosten für einen RAG-fähigen Anwendungsfall mit einem Amazon Kendra Kendra-Index, der 8.000 Interaktionen pro Tag unterstützt, belaufen sich auf etwa 204,26 USD pro Monat, zuzüglich der Kosten für die Wissensdatenbank.	Kosten

Wenn du willst.	Lesen.
Machen Sie sich mit den Sicherheitsüberleg ungen für diese Lösung vertraut.	Sicherheit
Erfahren Sie, wie Sie Kontingente für diese Lösung einplanen.	Kontingente
Erfahren Sie, welche AWS-Regionen diese Lösung unterstützen.	Unterstützte AWS-Regionen
Sehen Sie sich die in dieser Lösung enthalten e CloudFormation AWS-Vorlage an oder laden Sie sie herunter, um die Infrastrukturressourcen (den „Stack“) für diese Lösung automatisch bereitzustellen.	CloudFormation AWS-Vorlage
Greifen Sie auf den Quellcode zu und verwenden Sie optional das AWS Cloud Development Kit (AWS CDK), um die Lösung bereitzustellen.	GitHub Repository

Features und Vorteile

Die Generative AI Application Builder on AWS-Lösung bietet die folgenden Funktionen:

Schnelles Experimentieren

Mit dieser Lösung können Benutzer schnell experimentieren, da der Aufwand entfällt, der für die Bereitstellung mehrerer Instanzen mit unterschiedlichen Konfigurationen und für den Vergleich von Ergebnissen und Leistung erforderlich ist. Experimentieren Sie mit mehreren Konfigurationen LLMs, die verschiedene Parameter für schnelles Engineering, unternehmenseigene Wissensdatenbanken, Leitplanken, KI-Agenten und andere Parameter beinhalten.

Auswahl und Konfigurierbarkeit

Mit vorgefertigten Konnektoren für eine Vielzahl von Modellen LLMs, z. B. für Modelle, die über Amazon Bedrock erhältlich sind, bietet Ihnen diese Lösung die Flexibilität, das Modell Ihrer Wahl

sowie die von Ihnen bevorzugten AWS- und führenden FM-Services bereitzustellen. Sie können Amazon Bedrock Agents auch für die Erfüllung verschiedener Aufgaben und Workflows aktivieren.

Agent Builder

Erstellen und implementieren Sie produktionsbereite KI-Agenten mit vollständigem Lebenszyklusmanagement. Konfigurieren Sie Systemaufforderungen, integrieren Sie Model Context Protocol (MCP) -Server für Unternehmenstools und Datenzugriff und aktivieren Sie Speicherfunktionen für die Aufbewahrung des Kontextes über Konversationen hinweg. Agenten werden auf Amazon Bedrock AgentCore mit erweiterter Laufzeitunterstützung und Streaming-Antworten in Echtzeit bereitgestellt.

Workflow-Builder

Orchestrieren Sie mehrere Agent Builder-Agenten mithilfe der hierarchischen Delegierung zu komplexen Workflows. Erstellen Sie einen Supervisor-Agenten, der selbständig spezialisierte Agent Builder-Agenten auswählt und koordiniert, um mehrstufige Aufgaben zu erledigen. Konfigurieren Sie Agentenbeschreibungen, Delegierungsstrategien und Arbeitsspeicher auf Workflow-Ebene, während Sie bestehende Agent Builder-Bereitstellungen wiederverwenden.

Bereit für die Produktion

Diese Lösung basiert auf den Entwurfsprinzipien von AWS Well-Architected und bietet Sicherheit und Skalierbarkeit auf Unternehmensniveau mit hoher Verfügbarkeit und geringer Latenz. Dadurch wird eine nahtlose Integration in Ihre Anwendungen mit hohen Leistungsstandards gewährleistet.

Erweiterbare modulare Architektur

Erweitern Sie die Funktionalität dieser Lösung, indem Sie Ihre bestehenden Projekte integrieren oder zusätzliche AWS-Services nativ verbinden. Da es sich um eine Open-Source-Anwendung handelt, können Sie die mitgelieferte LangChain Orchestrierungsschicht oder Lambda-Funktionen verwenden, um eine Verbindung zu den Diensten Ihrer Wahl herzustellen.

Integration mit Service Catalog AppRegistry und Application Manager, einer Funktion von AWS Systems Manager

Diese Lösung umfasst eine Service AppRegistry Catalog-Ressource zur Registrierung der CloudFormation Lösungsvorlage und der zugrunde liegenden Ressourcen als Anwendung sowohl in AWS Service Catalog AppRegistry als auch im AWS Systems Manager Application Manager. Mit dieser Integration können Sie die Ressourcen der Lösung zentral verwalten.

Anwendungsfall Agent Builder im Vergleich zu Bedrock Agent

Diese Lösung bietet zwei unterschiedliche Ansätze für die Arbeit mit KI-Agenten, die jeweils für unterschiedliche Anwendungsfälle und Anforderungen geeignet sind:

Feature	Anwendungsfall Bedrock Agent	Agent Builder
Zweck	Rufen Sie vorab bereitgestellte Amazon Bedrock Agents auf	Erstellen, implementieren und verwalten Sie benutzerdefinierte Agenten
Konfiguration	Nur Agenten-ID und Alias-ID	Vollständige Agentenkonfiguration: Systemaufforderungen, Modelle, MCP-Server, Speicher
Bereitstellung	Einfache Aufrufebene	Vollständiger Agentenlebenszyklus auf Runtime AgentCore
Laufzeit	Amazon Bedrock Agentenservice	Amazon Bedrock AgentCore mit Strands SDK
Integration von Tools	In der Bedrock Agents-Konsole konfiguriert	Modellieren Sie Context Protocol (MCP) -Server und integrierte Strands-Tools
Arbeitsspeicher	Wird von Bedrock Agents verwaltet (bis zu 30 Tage)	AgentCore Speicher mit konfiguri erbarer Kurz- und Langzeits peicherung
Anpassung	Beschränkt auf vorab bereitgestellte Agenteneinstellungen	Volle Kontrolle über Eingabeau fforderungen, Modelle, Tools und Verhalten
Am besten geeignet für	Schnelle Bereitstellung vorhandener Agenten	Entwicklung kundenspezifischer Agenten und Produktionsbereits tellungen

> **Note**
>
> Beide Optionen unterstützen Echtzeit-Streaming, Konversationsverlauf und Sicherheit auf Unternehmensniveau.

Workflow-Builder

Workflow Builder ermöglicht die Orchestrierung mehrerer Agenten, indem ein Supervisor-Agent erstellt wird, der die Arbeit an spezialisierte Agent Builder-Agenten delegiert. Jeder Workflow besteht aus:

- Supervisor Agent: Der Einstiegsagent, der Benutzeranfragen entgegennimmt und spezialisierte Agenten koordiniert
- Spezialisierte Agenten: Agent Builder-Anwendungsfälle, an die der Supervisor Aufgaben delegieren kann
- Muster „Agenten als Tools“: Der Supervisor registriert jeden Agent Builder-Agenten als Tool und wählt selbstständig aus, welche Agenten verwendet werden sollen

Feature	Agent Builder	Workflow-Builder
Zweck	Erstellen und implementieren Sie einzelne benutzerdefinierte Agenten	Orchestrieren Sie mehrere Agent Builder-Agenten
Typ des Agenten	Einzelagent mit MCP-Tools	Supervisor Agent + mehrere Agent Builder-Agenten
Integration von Tools	MCP-Server und Strons-Tools	Agent Builder-Agenten, die als Tools registriert sind
Delegierung	Direkter Aufruf des Tools	Autonome Auswahl und Delegierung von Agenten
Komplexität	Aufgaben mit nur einem Agenten	Mehrstufige Workflows mit mehreren Agenten

Feature	Agent Builder	Workflow-Builder
Wiederverwendung von Agenten	–	Nutzt bestehende Agent Builder-Bereitstellungen wieder
Am besten geeignet für	Konzentrierte Aufgaben, die nur eine Domäne betreffen	Komplexe Workflows, die mehrere Spezialisierungen erfordern

Note

- Für Workflows ist mindestens ein Agent Builder-Anwendungsfall als spezialisierter Agent Agent erforderlich
- Bei allen spezialisierten Agenten muss es sich um Agent Builder-Anwendungsfälle handeln, die in GAAB bereitgestellt werden

Anwendungsfälle

Beantwortung von Fragen zu Unternehmensdaten

LLMs und andere Basismodelle wurden vorab anhand eines großen Datenkorpus trainiert, sodass sie viele Aufgaben der natürlichen Sprachverarbeitung (NLP) gut bewältigen können. Die meisten Basismodelle LLMs sind jedoch statisch und wurden vorab trainiert, was ihre Fähigkeit einschränkt, Fragen zu neuen, speziellen oder proprietären Themen präzise zu beantworten. Mithilfe von Lernaufforderungen können Sie die leistungsstarken NLP- und Textgenerierungsfunktionen eines LLMs nutzen, um Ihren Kunden ein umfassenderes Kundenerlebnis mit Ihren Unternehmensdaten zu bieten.

Schnelles generatives KI-Prototyping

Die Lösung ist standardmäßig mit verschiedenen Modellanbietern und Anwendungsfällen gebündelt. Mit einem benutzerfreundlichen Bereitstellungsassistenten können Kunden vorgefertigte Anwendungsfälle bereitstellen, um das schnelle Experimentieren mit verschiedenen generativen KI-Prototypen und -Workloads zu ermöglichen.

Vergleich und Erprobung mehrerer LLMs

LLMs arbeiten unterschiedlich, und angesichts der spezifischen Anforderungen Ihrer Anwendung stellen Sie möglicherweise fest, dass ein LLM besser zu Ihrer Anwendung passt als ein anderes. Dies kann auf Gründe zurückzuführen sein, die mit Leistung, Genauigkeit, Kosten, Kreativität oder vielen anderen Faktoren zusammenhängen. Mit dieser Lösung können Sie schnell mehrere Anwendungsfälle implementieren, sodass Sie mit verschiedenen Konfigurationen experimentieren und diese vergleichen können, bis Sie das gefunden haben, was Ihren Anforderungen entspricht.

Konzepte und Definitionen

In diesem Abschnitt werden die wichtigsten Konzepte beschrieben und die für diese Lösung spezifische Terminologie definiert:

Admin-Benutzer

Im Rahmen dieses Handbuchs ist der Admin-Benutzer für die Verwaltung der in der Bereitstellung enthaltenen Inhalte verantwortlich. Dieser Benutzer erhält Zugriff auf die Benutzeroberfläche des Deployment-Dashboards und ist in erster Linie für die Pflege der Benutzererfahrung in Unternehmen verantwortlich. Dies ist unser primärer Zielkunde.

gewerblicher Nutzer

Im Rahmen dieses Handbuchs steht der Geschäftsbenutzer für die Personen, für die der Anwendungsfall bereitgestellt wurde. Sie sind die Nutzer der Wissensdatenbank und der Kunde, der für die Bewertung und das Experimentieren mit der LLMs Wissensdatenbank verantwortlich ist.

Bereitstellungs-Dashboard

Das Deployment-Dashboard ist eine Weboberfläche, die als Verwaltungskonsole für Administratorbenutzer dient, um ihre Anwendungsfälle anzusehen, zu verwalten und zu erstellen. Dieses Dashboard ermöglicht es Kunden, schnell zu experimentieren, zu iterieren und verschiedene AI/ML Workloads zu nutzen. LLMs

DevOps user

Im Rahmen dieses Handbuchs ist der DevOps Benutzer für die Bereitstellung der Lösung innerhalb des AWS-Kontos und für die Verwaltung der Infrastruktur, die Aktualisierung der Lösung, die Überwachung der Leistung und die Aufrechterhaltung des allgemeinen Zustands und Lebenszyklus der Lösung verantwortlich.

Anwendungsfall

Anwendungsfälle sind isolierte Anwendungen aus der Gesamtlösung, die in die Gesamtlösung integriert werden LLMs , um ein umfassenderes Kundenerlebnis zu ermöglichen, indem neue oder bestehende Anwendungen um eine Benutzeroberfläche in natürlicher Sprache erweitert werden können. Anwendungsfälle können über das Deployment-Dashboard oder eigenständig bereitgestellt werden.

> **Note**
>
> Eine allgemeine Referenz zu AWS-Begriffen finden Sie im AWS-Glossar.

Übersicht über die Architektur

Dieser Abschnitt enthält Referenzdiagramme zur Implementierungsarchitektur für die mit dieser Lösung bereitgestellten Komponenten.

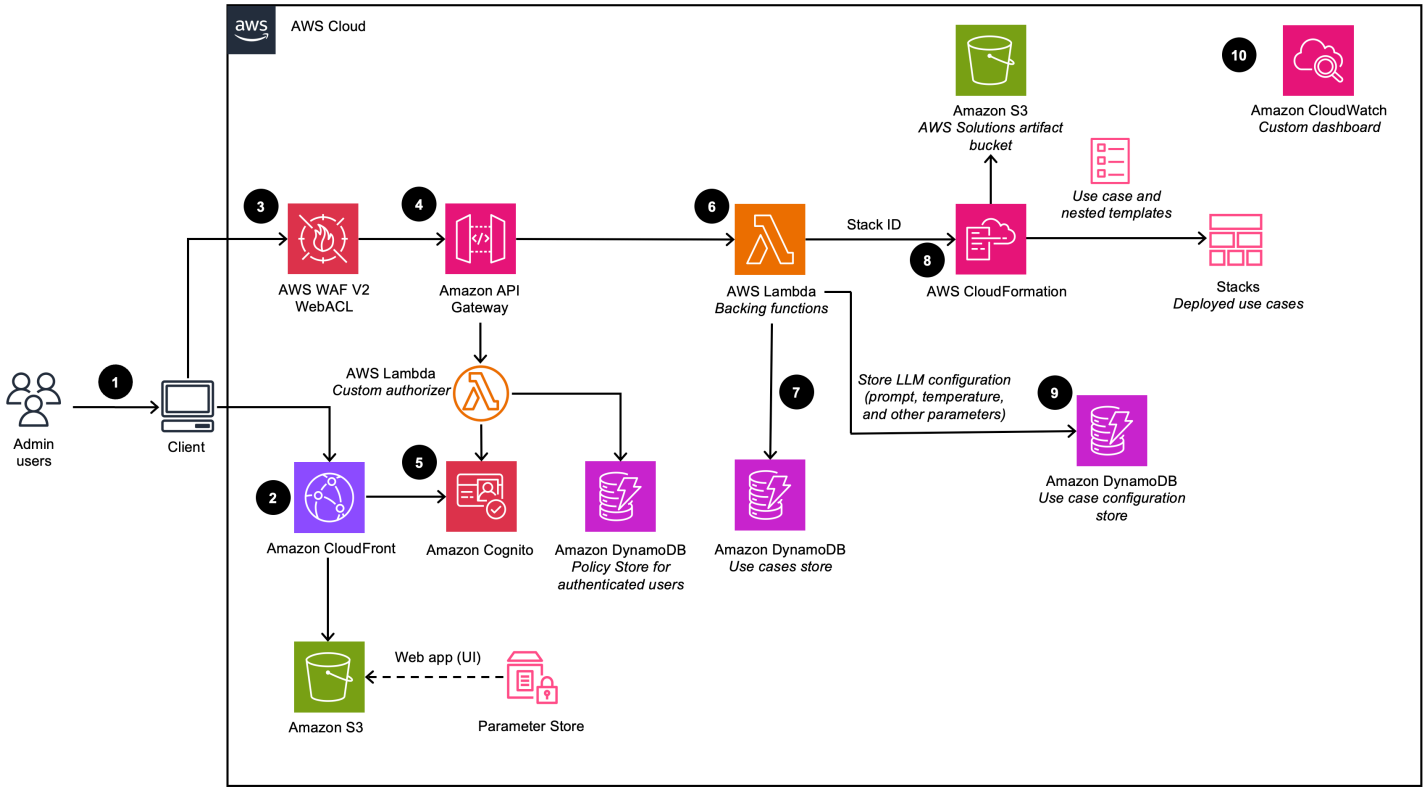
Architekturdiagramme

Um mehrere Anwendungsfälle und Geschäftsanforderungen zu unterstützen, bietet diese Lösung sechs CloudFormation AWS-Vorlagen:

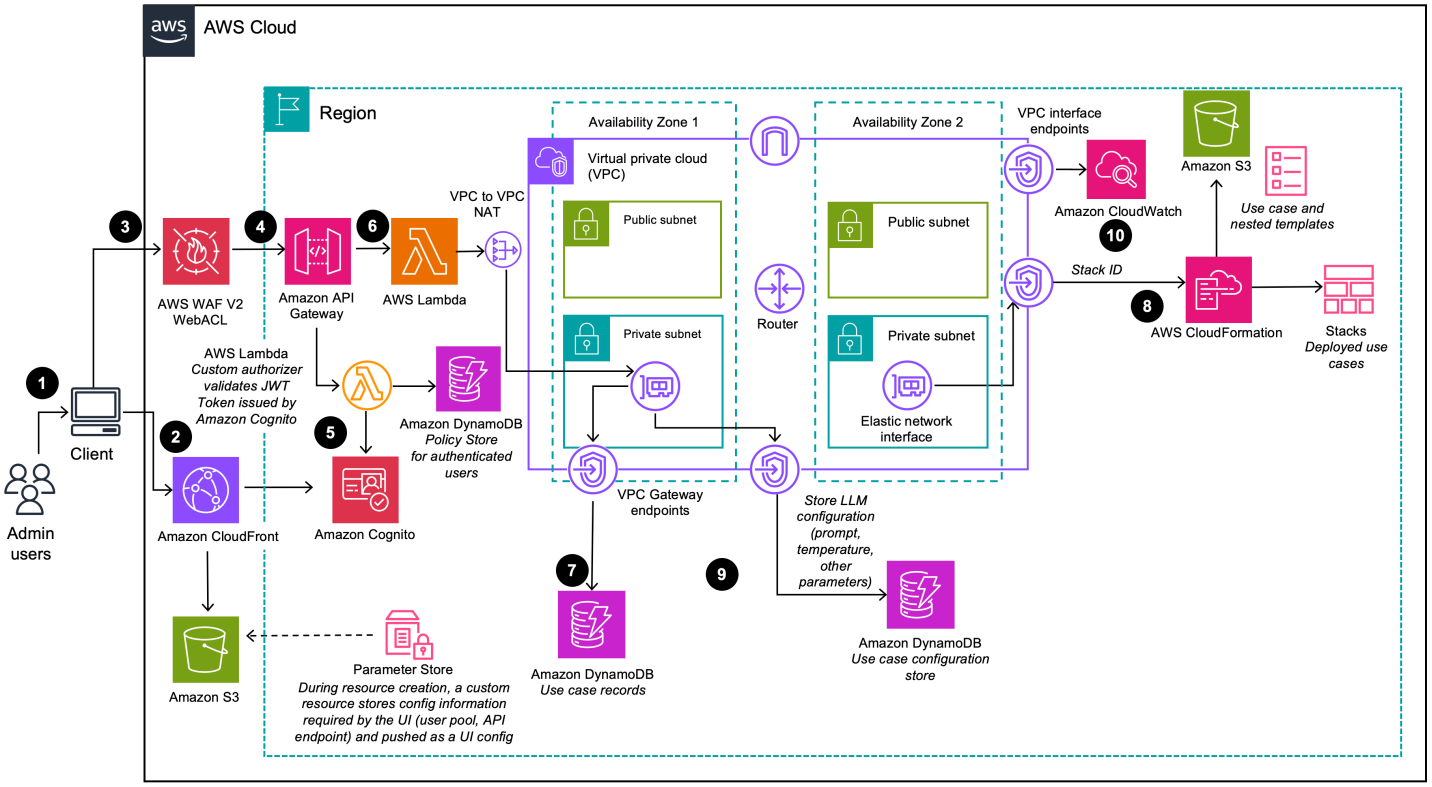
1. Bereitstellungs-Dashboard — Das Bereitstellungs-Dashboard ist eine Weboberfläche, die als Verwaltungskonsole für Administratorbenutzer dient, um ihre Anwendungsfälle anzusehen, zu verwalten und zu erstellen. Dieses Dashboard ermöglicht es Kunden, verschiedene AI/ML Workloads schnell zu testen, zu iterieren und in der Produktion zu nutzen. LLMs
2. Text-Anwendungsfall — Der Text-Anwendungsfall ermöglicht es Benutzern, eine Benutzeroberfläche in natürlicher Sprache mithilfe generativer KI zu erleben. Dieser Anwendungsfall kann in neue oder bestehende Anwendungen integriert werden und kann über das Deployment-Dashboard oder unabhängig über eine bereitgestellte URL bereitgestellt werden.
3. Anwendungsfall Bedrock Agent — Der Bedrock Agent-Anwendungsfall ermöglicht die Verwendung vorhandener Bedrock Agents, um Aufgaben zu erledigen oder sich wiederholende Workflows zu automatisieren.
4. MCP Server — Der MCP Server-Anwendungsfall ermöglicht die Bereitstellung und Verwaltung von Model Context Protocol-Servern, die einen standardisierten Tool- und Ressourcenzugriff für KI-Anwendungen bieten. Unterstützt sowohl Gateway-Methoden zum Umschließen vorhandener Lambda-Funktionen und externer MCP-Server als auch Laufzeitmethoden für die Bereitstellung von benutzerdefinierten containerisierten MCP-Servern. APIs
5. Agent Builder — Der Agent Builder ermöglicht die Erstellung und Bereitstellung von produktionsbereiten KI-Agenten auf Amazon Bedrock AgentCore mit vollständiger Konfigurationskontrolle, MCP-Serverintegration und Speicherverwaltungsfunktionen.
6. Workflow Builder — Der Workflow Builder ermöglicht die Erstellung von Supervisor-Agenten, die mehrere Agent Builder-Agenten mithilfe des Delegationsmusters Agents as Tools für komplexe Workflows mit mehreren Agenten orchestrieren.

Bereitstellungs-Dashboard

Zeigt die Architektur des Bereitstellungs-Dashboards (bei Bereitstellung mit deaktivierter VPC-Option)



Zeigt die Architektur des Bereitstellungs-Dashboards (bei Bereitstellung mit aktivierter VPC-Option)



> **Note**
>
> CloudFormation AWS-Ressourcen werden aus Konstrukten des AWS Cloud Development Kit (AWS CDK) erstellt.

Der allgemeine Prozessablauf für die mit der CloudFormation AWS-Vorlage bereitgestellten Lösungskomponenten sieht wie folgt aus:

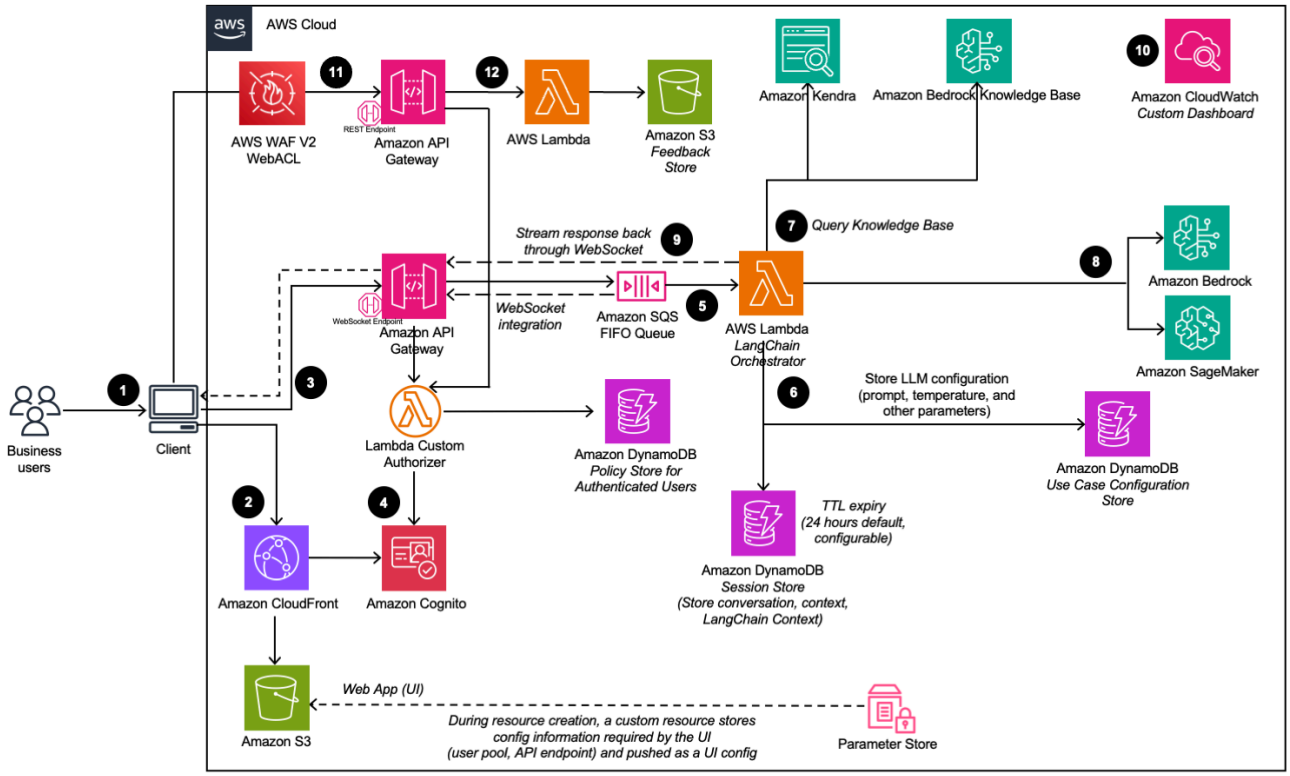
1. Admin-Benutzer melden sich bei der Benutzeroberfläche (UI) des Deployment Dashboards an.
2. Amazon CloudFront stellt die Web-Benutzeroberfläche bereit, die in einem Amazon Simple Storage Service (Amazon S3) -Bucket gehostet wird.
3. AWS WAF schützt sie vor APIs Angriffen. Diese Lösung konfiguriert eine Reihe von Regeln, die als Web Access Control List (Web ACL) bezeichnet werden und Webanfragen auf der Grundlage konfigurierbarer, benutzerdefinierter Websicherheitsregeln und -bedingungen zulassen, blockieren oder zählen.
4. Die Weboberfläche nutzt eine Reihe von REST APIs , die über Amazon API Gateway verfügbar gemacht werden.

5. Amazon Cognito authentifiziert Benutzer und unterstützt sowohl die CloudFront Web-Benutzeroberfläche als auch das API Gateway.
6. AWS Lambda stellt die Geschäftslogik für die REST-Endpunkte bereit. Diese unterstützende Lambda-Funktion verwaltet und erstellt die erforderlichen Ressourcen für die Durchführung von Anwendungsfallbereitstellungen mit AWS. CloudFormation
7. Amazon DynamoDB speichert die Liste der Bereitstellungen.
8. Wenn ein neuer Anwendungsfall vom Admin-Benutzer erstellt wird, initiiert die unterstützende Lambda-Funktion ein CloudFormation Stack-Erstellungsereignis für den angeforderten Anwendungsfall.
9. Alle vom Admin-Benutzer im Einrichtungsassistenten bereitgestellten LLM-Konfigurationsoptionen werden in DynamoDB gespeichert. Die Bereitstellung verwendet diese DynamoDB-Tabelle, um das LLM zur Laufzeit zu konfigurieren.
10. Mithilfe von Amazon CloudWatch sammelt diese Lösung Betriebsmetriken von verschiedenen Diensten, um benutzerdefinierte Dashboards zu generieren, mit denen Sie die Leistung und den Betriebsstatus der Lösung überwachen können.

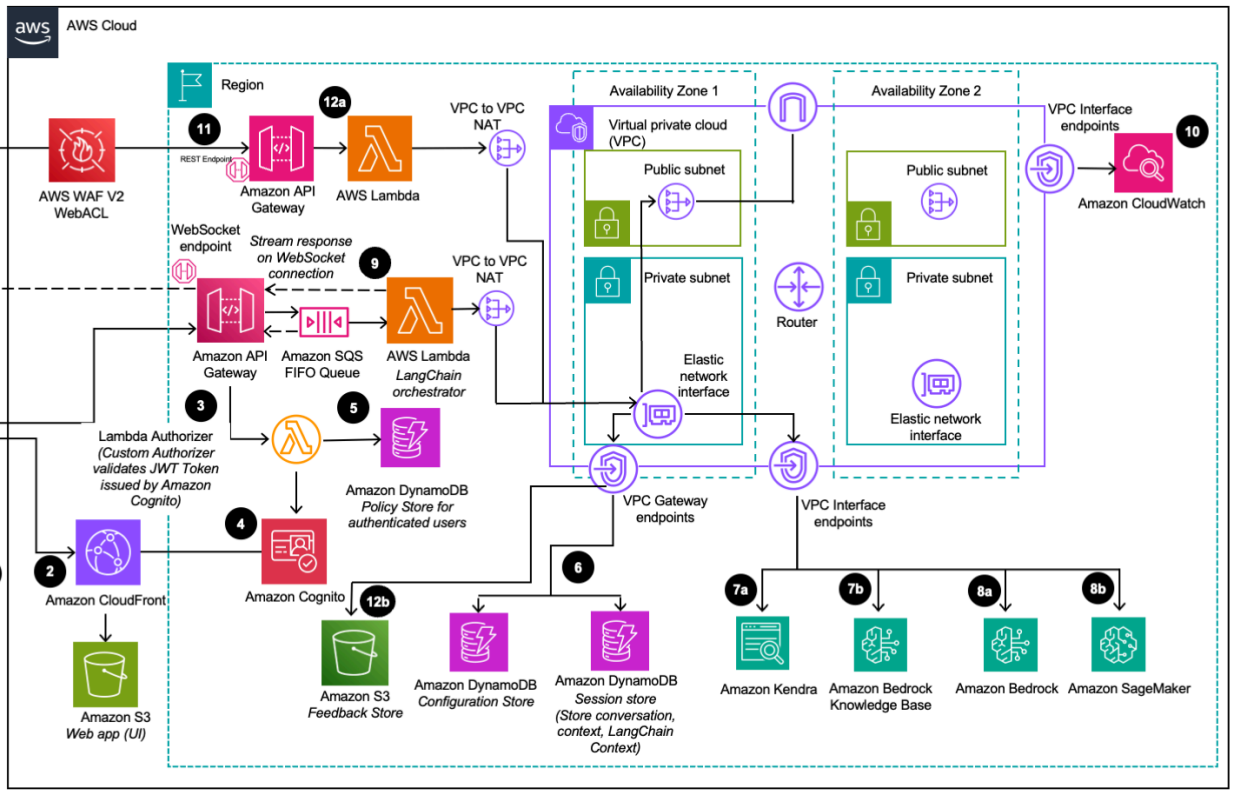
> **Note**
>
> - Wenn Sie sich dafür entscheiden, diese Lösung in einer Amazon VPC bereitzustellen, werden die Daten innerhalb Ihres privaten Netzwerks weitergeleitet.
> - Obwohl das Deployment-Dashboard in den meisten AWS-Regionen gestartet werden kann, unterliegen die bereitgestellten Anwendungsfälle bestimmten Einschränkungen, die von der Verfügbarkeit der Services abhängen. Weitere Informationen finden Sie unter Unterstützte AWS-Regionen.

Anwendungsfall im Textformat

Stellt die Architektur des Text-Anwendungsfalls dar (bei der Bereitstellung mit deaktivierter VPC-Option)



Stellt die Architektur des Text-Anwendungsfalls dar (bei Bereitstellung mit aktivierter VPC-Option)



Der allgemeine Prozessablauf für die mit der CloudFormation AWS-Vorlage bereitgestellten Lösungskomponenten sieht wie folgt aus:

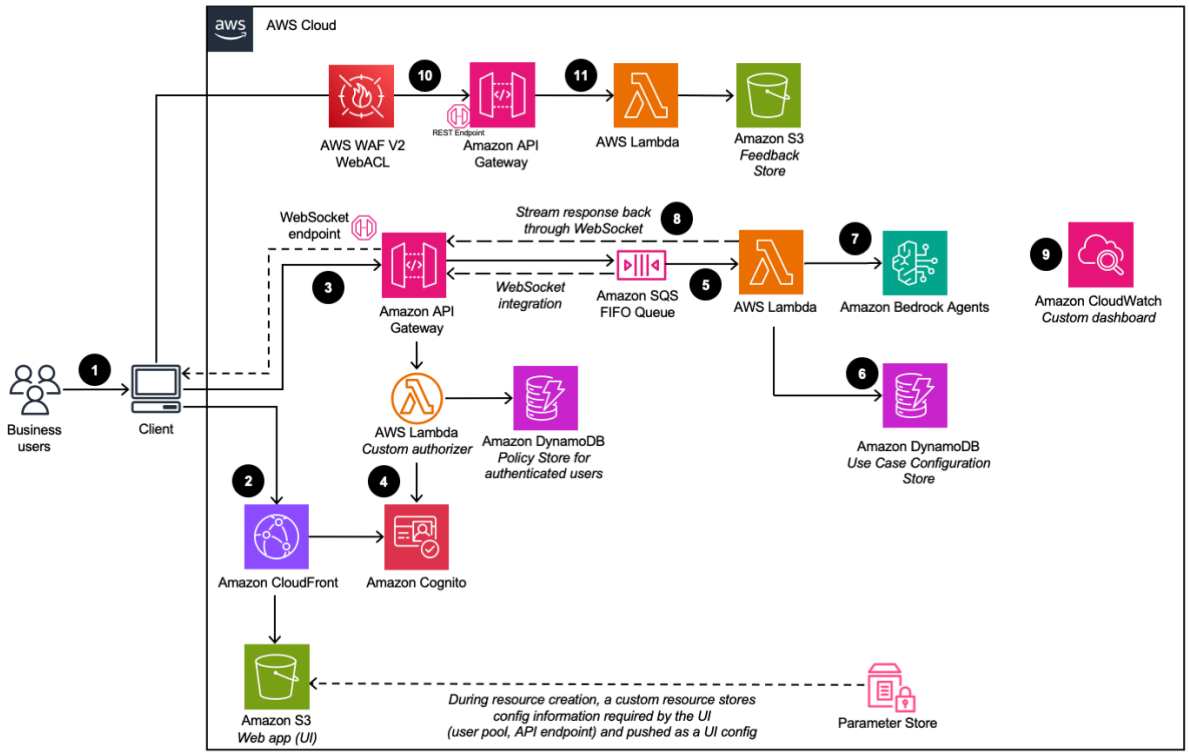
1. Admin-Benutzer stellen den Anwendungsfall mithilfe des Deployment Dashboards bereit. Geschäftsbenutzer melden sich bei der Benutzeroberfläche für Anwendungsfälle an.
2. CloudFront stellt die Web-Benutzeroberfläche bereit, die in einem S3-Bucket gehostet wird.
3. Die Weboberfläche nutzt eine WebSocket Integration, die mit API Gateway erstellt wurde. Das API Gateway wird von einer benutzerdefinierten Lambda-Autorisierungsfunktion unterstützt, die die entsprechende AWS Identity and Access Management (IAM) -Richtlinie zurückgibt, die auf der Amazon Cognito Cognito-Gruppe basiert, zu der der authentifizierende Benutzer gehört. Die Richtlinie ist in DynamoDB gespeichert.
4. Amazon Cognito authentifiziert Benutzer und unterstützt sowohl die CloudFront Web-Benutzeroberfläche als auch das API Gateway.
5. Eingehende Anfragen des Geschäftsbenutzers werden vom API Gateway an eine Amazon SQS SQS-Warteschlange und dann an den LangChain Orchestrator weitergeleitet. Der LangChain Orchestrator ist eine Sammlung von Lambda-Funktionen und -Ebenen, die die Geschäftslogik für die Erfüllung von Anfragen von Geschäftsbenutzern bereitstellen. Die Warteschlange ermöglicht den asynchronen Betrieb der API-Gateway-Lambda-Integration. Die Warteschlange leitet Verbindungsinformationen an die Lambda-Funktionen weiter, die dann die Ergebnisse direkt an die API Gateway Gateway-Websocket-Verbindung zurücksenden, um lang andauernde Inferenzrufe zu unterstützen.
6. Der LangChain Orchestrator verwendet Amazon DynamoDB, um die konfigurierten LLM-Optionen und die erforderlichen Sitzungsinformationen (wie den Chat-Verlauf) abzurufen.
7. Wenn für die Bereitstellung eine Wissensdatenbank aktiviert ist, nutzt der LangChain Orchestrator Amazon Kendra oder Knowledge Bases for Amazon Bedrock, um eine Suchabfrage zum Abrufen von Dokumentauszügen auszuführen.
8. Mithilfe des Chat-Verlaufs, der Abfrage und des Kontextes aus der Wissensdatenbank erstellt der LangChain Orchestrator die endgültige Aufforderung und sendet die Anfrage an das LLM, das auf Amazon Bedrock oder Amazon AI gehostet wird. SageMaker
9. Wenn die Antwort vom LLM zurückkommt, streamt der LangChain Orchestrator die Antwort zurück über das API Gateway, WebSocket damit sie von der Client-Anwendung verarbeitet wird.
10. Mithilfe von Amazon CloudWatch sammelt diese Lösung Betriebsmetriken von verschiedenen Diensten, um benutzerdefinierte Dashboards zu generieren, mit denen Sie die Leistung und den Betriebsstatus der Bereitstellung überwachen können.

11. Wenn die Erfassung von Feedback aktiviert ist, wird ein REST-API-Endpunkt, der Amazon API Gateway nutzt, für die Erfassung von Benutzerfeedback zur Verfügung gestellt.
12. Das Feedback, das Lambda unterstützt, erweitert das übermittelte Feedback um zusätzliche anwendungsfallspezifische Metadaten (z. B. das verwendete Modell) und speichert die Daten in Amazon S3 für spätere Analysen und Berichte durch die DevOps Benutzer.

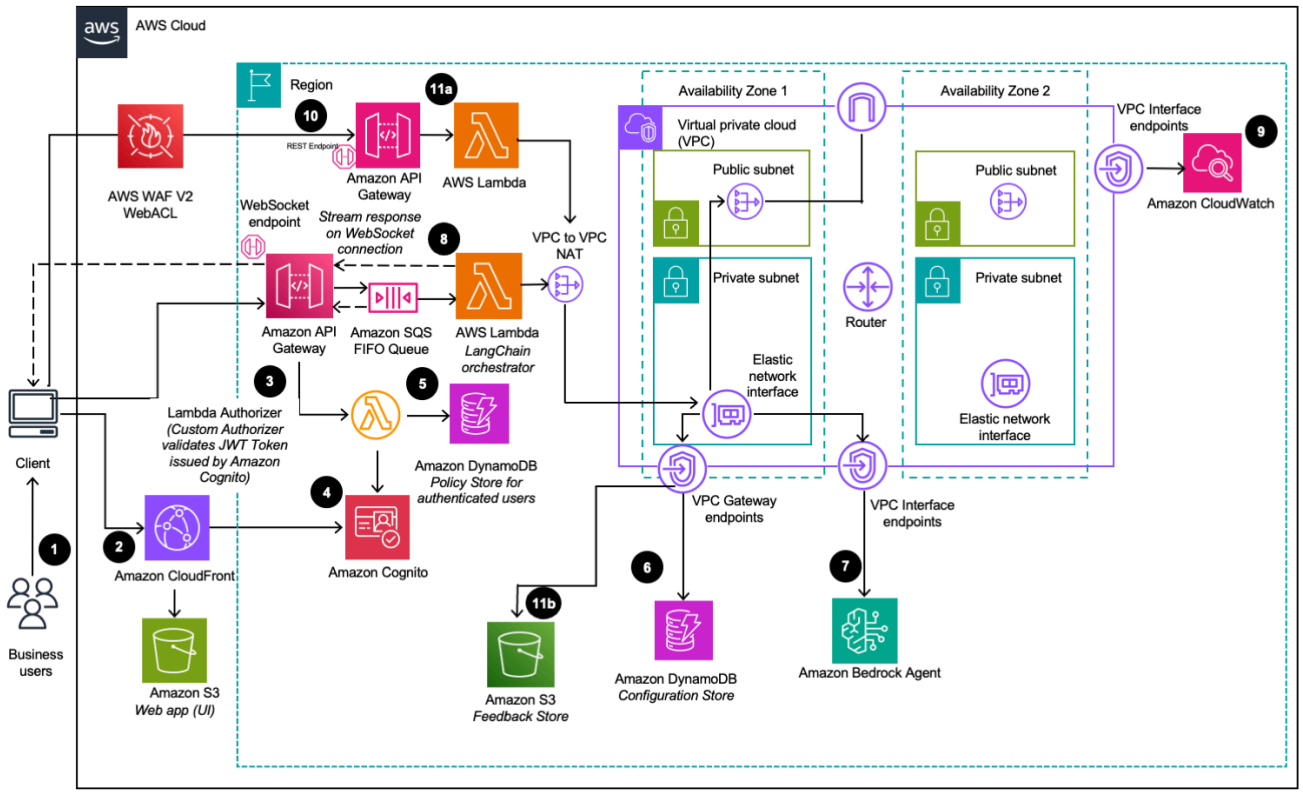
> **Note**
>
> Wenn Sie sich dafür entscheiden, diese Lösung in einer Amazon VPC bereitzustellen, werden die Daten an Ihr privates Netzwerk weitergeleitet.

Anwendungsfall Bedrock Agent

Stellt die Bedrock Agent-Anwendungsfallarchitektur dar (bei Bereitstellung mit deaktivierter VPC-Option)



Stellt die Bedrock Agent-Anwendungsfallarchitektur dar (bei Bereitstellung mit aktivierter VPC-Option)



Der allgemeine Prozessablauf für die mit der CloudFormation AWS-Vorlage bereitgestellten Lösungskomponenten sieht wie folgt aus:

1. Admin-Benutzer stellen den Anwendungsfall mithilfe des Deployment Dashboards bereit. Geschäftsanwender melden sich bei der Benutzeroberfläche für Anwendungsfälle an.
2. CloudFront stellt die Webbenutzeroberfläche bereit, die in einem S3-Bucket gehostet wird.
3. Die Weboberfläche nutzt eine WebSocket Integration, die mit API Gateway erstellt wurde. Das API Gateway wird von einer benutzerdefinierten Lambda-Autorisierungsfunktion unterstützt, die die entsprechende AWS Identity and Access Management (IAM) -Richtlinie zurückgibt, die auf der Amazon Cognito Cognito-Gruppe basiert, zu der der authentifizierende Benutzer gehört. Die Richtlinie ist in DynamoDB gespeichert.
4. Amazon Cognito authentifiziert Benutzer und unterstützt sowohl die CloudFront Web-Benutzeroberfläche als auch das API Gateway.
5. Eingehende Anfragen des Geschäftsbenutzers werden von API Gateway an eine Amazon SQS SQS-Warteschlange und dann an die AWS Lambda Lambda-Funktion weitergeleitet. Die Warteschlange ermöglicht den asynchronen Betrieb der API-Gateway-Lambda-Integration. Die Warteschlange leitet Verbindungsinformationen an die Lambda-Funktion weiter, die dann die

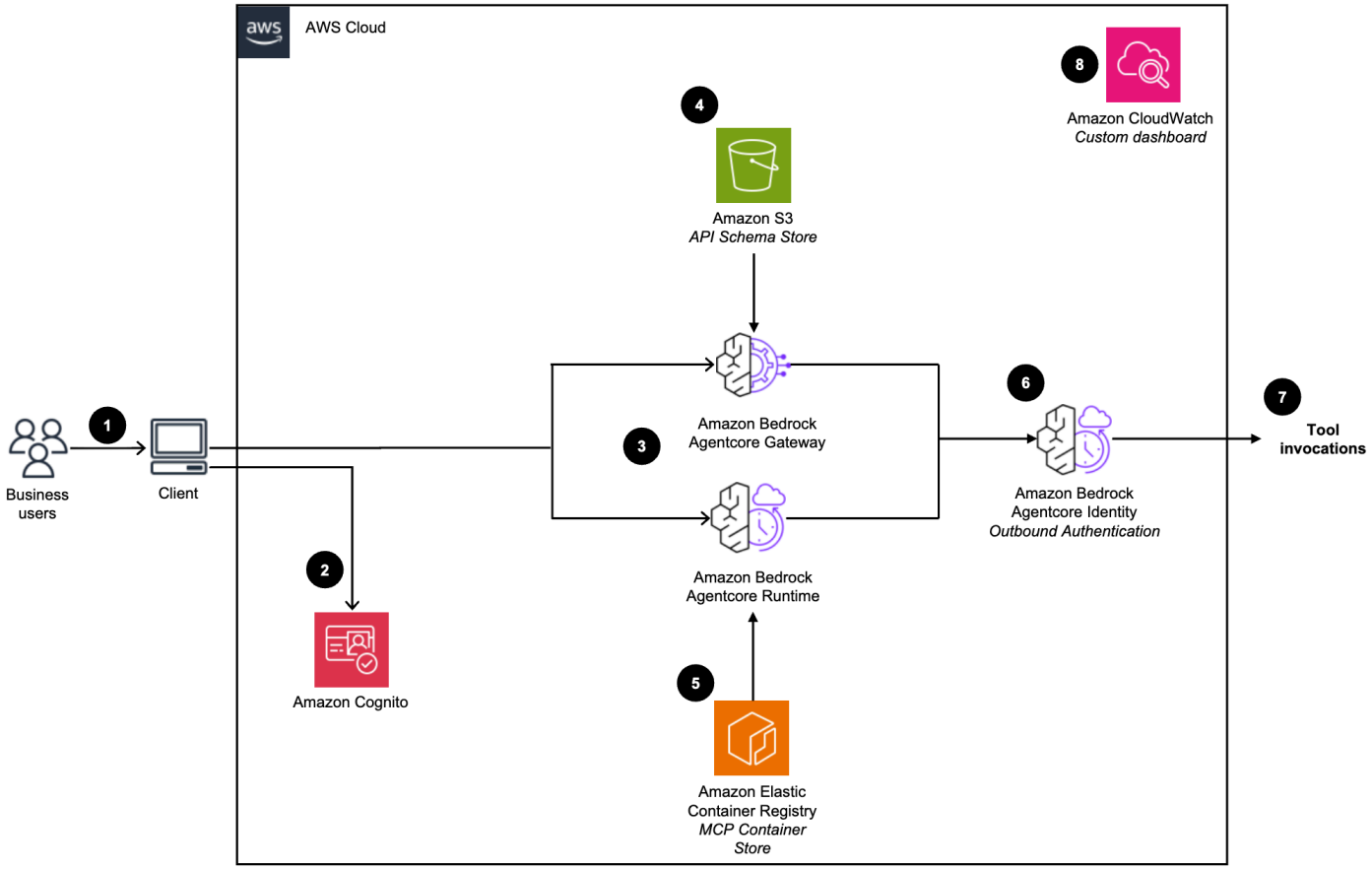
Ergebnisse direkt an die API Gateway Gateway-Websocket-Verbindung zurücksendet, um lang andauernde Inferenzrufe zu unterstützen.

6. Die AWS Lambda Lambda-Funktion verwendet Amazon DynamoDB, um die Anwendungsfallkonfigurationen nach Bedarf abzurufen.
7. Unter Verwendung der Benutzereingabe und aller relevanten Anwendungsfallkonfigurationen erstellt die AWS Lambda Lambda-Funktion eine Anforderungs-Payload und sendet sie an den konfigurierten Amazon Bedrock Agent, um die Benutzerabsicht zu erfüllen.
8. Wenn die Antwort vom Amazon Bedrock Agent zurückkommt, streamt die Lambda-Funktion die Antwort zurück über das API Gateway, WebSocket damit sie von der Client-Anwendung verarbeitet wird.
9. Mithilfe von Amazon CloudWatch sammelt diese Lösung Betriebsmetriken von verschiedenen Diensten, um benutzerdefinierte Dashboards zu generieren, mit denen Sie die Leistung und den Betriebsstatus der Bereitstellung überwachen können.
10. Wenn die Erfassung von Feedback aktiviert ist, wird ein REST-API-Endpunkt, der Amazon API Gateway nutzt, für die Erfassung von Benutzerfeedback zur Verfügung gestellt.
11. Das Feedback, das Lambda unterstützt, erweitert das übermittelte Feedback um zusätzliche anwendungsfallspezifische Metadaten und speichert die Daten in Amazon S3 für spätere Analysen und Berichte durch die DevOps Benutzer.

> **Note**
>
> Wenn Sie sich dafür entscheiden, diese Lösung in einer Amazon VPC bereitzustellen, werden Daten innerhalb Ihres privaten Netzwerks weitergeleitet.

Anwendungsfall für MCP-Server

Stellt die Architektur des MCP Server-Anwendungsfalls dar



Der MCP Server-Anwendungsfall ermöglicht die Bereitstellung und Verwaltung von Model Context Protocol-Servern auf Amazon AgentCore Bedrock. MCP-Server bieten eine standardisierte Schnittstelle für KI-Anwendungen für den Zugriff auf Tools, Ressourcen und Unternehmensdatenquellen.

Die Lösung unterstützt zwei Bereitstellungsmethoden:

- Gateway-Methode: Schließt bestehende Lambda-Funktionen APIs, REST oder externe MCP-Server als MCP-Tools ein und übernimmt die Protokollübersetzung automatisch
- Runtime-Methode: Stellt benutzerdefinierte containerisierte MCP-Server aus Amazon ECR-Images bereit

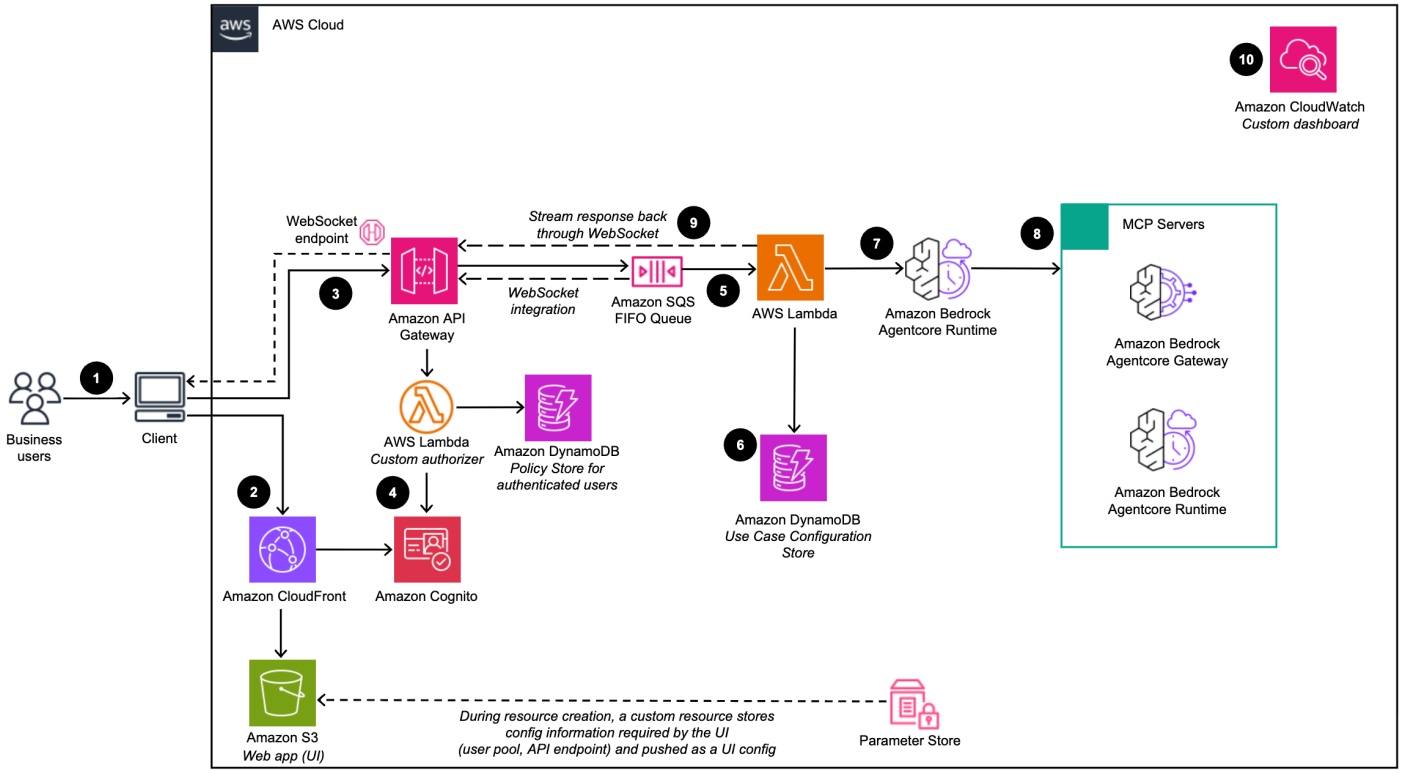
Der allgemeine Prozessablauf für die Bereitstellung von MCP-Servern sieht wie folgt aus:

1. Admin-Benutzer stellen den MCP-Server-Anwendungsfall mithilfe des Deployment Dashboards bereit und wählen entweder die Gateway- oder Runtime-Bereitstellungsmethode aus.

2. Diese Aktion ist mit Amazon Cognito authentifiziert.
3. Für die Gateway-Bereitstellung erstellt die Lösung ein Amazon Bedrock AgentCore Gateway, das bestehende Lambda-Funktionen oder externe MCP-Server in MCP-konforme Tools umwandelt. APIs Für die Runtime-Bereitstellung stellt die Lösung containerisierte MCP-Server auf Amazon Bedrock AgentCore Runtime mithilfe der bereitgestellten ECR-Images bereit.
4. Gateway-Bereitstellungen rufen die erforderlichen API/Lambda/Smithy Schemas von ihrem hochgeladenen Speicherort in Amazon S3 ab oder stellen eine direkte Verbindung zu MCP-Server-URL-Endpunkten her.
5. Runtime-Bereitstellungen rufen den vom Benutzer bereitgestellten containerisierten MCP-Server aus der Amazon Elastic Container Registry (ECR) ab
6. Der MCP-Server ist mit einem Amazon AgentCore Bedrock Identity-Client ausgestattet OAuth
7. Der MCP-Server stellt die zugehörigen Tools am /mcp-Endpunkt zur Verfügung, damit Agenten sie entdecken können.
8. Amazon CloudWatch sammelt Betriebsmetriken und Protokolle von MCP-Serverbereitstellungen zur Überwachung und Fehlerbehebung.

Anwendungsfall Agent Builder

Stellt die Agent Builder-Architektur dar



Der allgemeine Prozessablauf für die mit der CloudFormation AWS-Vorlage bereitgestellten Agent Builder-Komponenten sieht wie folgt aus:

1. Admin-Benutzer stellen den Anwendungsfall mithilfe des Deployment Dashboards bereit. Geschäftsanwender melden sich bei der Benutzeroberfläche für Anwendungsfälle an.
2. CloudFront stellt die Webbenutzeroberfläche bereit, die in einem S3-Bucket gehostet wird.
3. Die Weboberfläche nutzt eine WebSocket Integration, die mit API Gateway erstellt wurde. Das API Gateway wird von einer benutzerdefinierten Lambda-Autorisierungsfunktion unterstützt, die die entsprechende AWS Identity and Access Management (IAM) -Richtlinie zurückgibt, die auf der Amazon Cognito Cognito-Gruppe basiert, zu der der authentifizierende Benutzer gehört. Die Richtlinie ist in DynamoDB gespeichert.
4. Amazon Cognito authentifiziert Benutzer und unterstützt sowohl die CloudFront Web-Benutzeroberfläche als auch das API Gateway.
5. Eingehende Anfragen des Geschäftsbenutzers werden von API Gateway an eine Amazon SQS SQS-Warteschlange und dann an die AWS Lambda Lambda-Funktion weitergeleitet. Die Warteschlange ermöglicht den asynchronen Betrieb der API-Gateway-Lambda-Integration. Die Warteschlange leitet Verbindungsinformationen an die Lambda-Funktion weiter, die dann die

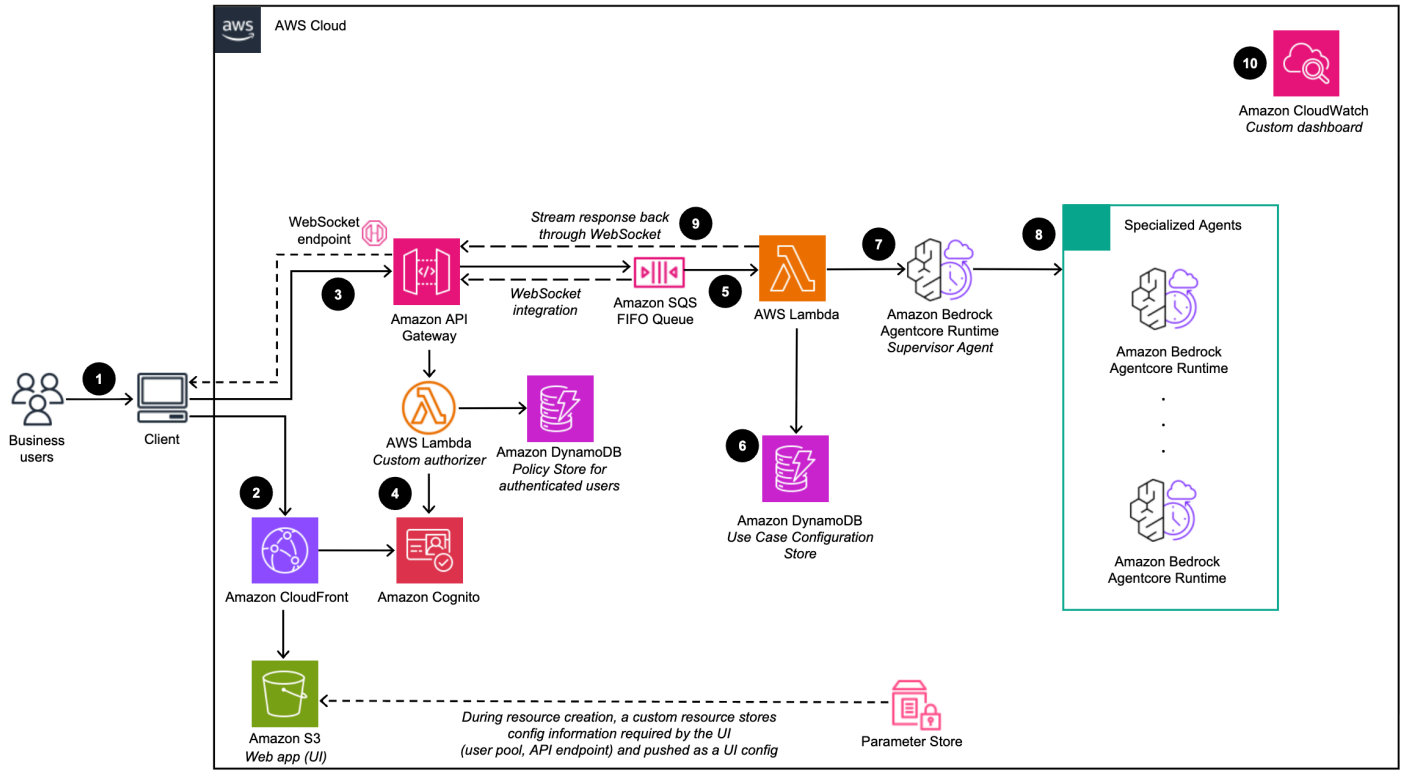
Ergebnisse direkt an die API Gateway Gateway-Websocket-Verbindung zurücksendet, um lang andauernde Inferenzrufe zu unterstützen.

6. Die AWS Lambda Lambda-Funktion ruft die Agentenkonfiguration von DynamoDB ab.
7. Unter Verwendung der Benutzereingaben und aller relevanten Anwendungsfallkonfigurationen erstellt die AWS Lambda Lambda-Funktion eine Anforderungs-Payload und sendet sie an den Agenten, der auf Amazon AgentCore Bedrock Runtime ausgeführt wird.
8. Der Agent stellt eine Verbindung zu den zugehörigen MCP-Servern her und registriert die Tools auf der Strings-Agenten-Instance. Der Agent wählt dann selbstständig Aktionen auf der Grundlage von Toolbeschreibungen und Aufgabenanforderungen aus und führt sie aus.
9. Wenn die Antwort von der Amazon AgentCore Bedrock-Laufzeit zurückkommt, streamt die Lambda-Funktion die Antwort zurück über das API Gateway, WebSocket damit sie von der Client-Anwendung verarbeitet wird.

> **Note**
>
> - Die Agentenverarbeitung ist auf das Lambda-Ausführungstimeout (15 Minuten) beschränkt.

Anwendungsfall Workflow Builder

Stellt die Workflow Builder-Architektur dar



Der allgemeine Prozessablauf für die mit der CloudFormation AWS-Vorlage bereitgestellten Workflow Builder-Komponenten sieht wie folgt aus:

1. Admin-Benutzer stellen den Workflow mithilfe des Deployment Dashboards bereit und wählen Agent Builder-Agenten aus, die als spezialisierte Agenten aufgenommen werden sollen.
2. CloudFront stellt die Weboberfläche bereit, die in einem S3-Bucket gehostet wird.
3. Die Weboberfläche nutzt eine WebSocket Integration, die mit API Gateway erstellt wurde. Das API Gateway wird von einer benutzerdefinierten Lambda-Autorisierungsfunktion unterstützt, die die entsprechende AWS Identity and Access Management (IAM) -Richtlinie zurückgibt, die auf der Amazon Cognito Cognito-Gruppe basiert, zu der der authentifizierende Benutzer gehört. Die Richtlinie ist in DynamoDB gespeichert.
4. Amazon Cognito authentifiziert Benutzer und unterstützt sowohl die CloudFront Web-Benutzeroberfläche als auch das API Gateway.
5. Eingehende Anfragen des Geschäftsbenutzers werden von API Gateway an eine Amazon SQS SQS-Warteschlange und dann an die AWS Lambda Lambda-Funktion weitergeleitet. Die Warteschlange ermöglicht den asynchronen Betrieb der API-Gateway-Lambda-Integration.
6. Die AWS Lambda Lambda-Funktion ruft die Workflow-Konfiguration von DynamoDB ab, einschließlich der Liste der spezialisierten Agent Builder-Agenten.

7. Mithilfe der Benutzereingabe und der Workflow-Konfiguration sendet Lambda Anfragen an die Amazon Bedrock AgentCore Runtime, die den Supervisor Agent hostet.
8. Der Supervisor-Agent erstellt lokale Instanzen aller spezialisierten Agent Builder-Agenten innerhalb der AgentCore Runtime-Umgebung. Diese spezialisierten Agenten werden mithilfe des Patterns Agents as Tools als Tools als Tools registriert. Der Supervisor wählt dann selbstständig die Arbeit aus und delegiert sie auf der Grundlage der Agentenbeschreibungen und Aufgabenanforderungen an spezialisierte Agenten.
9. Der Supervisor-Agent aggregiert die Ergebnisse spezialisierter Agenten und formuliert die endgültige Antwort. Er gibt sie an Lambda zurück, damit sie über den API-Gateway-Websocket zurück an die Client-Anwendung gestreamt wird.

> **Note**
>
> - Die Workflow-Verarbeitung ist auf das Lambda-Ausführungstimeout (15 Minuten) beschränkt.

Überlegungen zum AWS-Well-Architected-Design

Diese Lösung wurde mit Best Practices aus dem AWS Well-Architected Framework entwickelt, das Kunden dabei unterstützt, zuverlässige, sichere, effiziente und kostengünstige Workloads in der Cloud zu entwerfen und zu betreiben.

In diesem Abschnitt wird beschrieben, wie die Entwurfsprinzipien und Best Practices des Well-Architected Framework bei der Erstellung dieser Lösung angewendet wurden.

Operative Exzellenz

In diesem Abschnitt wird beschrieben, wie wir diese Lösung unter Verwendung der Prinzipien und bewährten Verfahren des Pfeilers Operational Excellence konzipiert haben.

- Wir haben die Lösung so entwickelt, als infrastructure-as-code würden wir Amazon verwenden CloudFormation.
- Lambda-Funktionen übertragen benutzerdefinierte Metriken CloudWatch und ein benutzerdefiniertes CloudWatch Dashboard, um den Zustand der Lösung zu überwachen.

- Die Lösungskomponenten sind stark modularisiert, sodass Sie flexibel auswählen können, welche Komponenten bereitgestellt werden sollen.

Sicherheit

In diesem Abschnitt wird beschrieben, wie wir diese Lösung unter Verwendung der Prinzipien und bewährten Verfahren der Sicherheitssäule konzipiert haben.

- Das Deployment-Dashboard und alle Anwendungsfälle sind bei Amazon Cognito authentifiziert und autorisiert.
- Die gesamte dienstübergreifende Kommunikation verwendet AWS IAM-Rollen.
- Alle Lösungsrollen folgen dem Zugriff mit den geringsten Rechten, d. h., es werden nur die erforderlichen Mindestberechtigungen gewährt.
- Alle Datenspeicher, einschließlich S3-Buckets, DynamoDB und Amazon Kendra, verfügen über Verschlüsselung im Ruhezustand.

Zuverlässigkeit

In diesem Abschnitt wird beschrieben, wie wir diese Lösung unter Verwendung der Prinzipien und bewährten Verfahren der Zuverlässigkeitskomponente konzipiert haben.

- Architektur, die auf dem serverlosen Paradigma basiert.
- Wir haben die Architektur für horizontale Skalierbarkeit auf Abruf und automatische Wiederherstellung nach einem Ausfall der zugrunde liegenden Infrastruktur entwickelt.
- Die Architektur umfasst Pufferung und Drosselung von Anfragen, um die zugrunde liegenden Endgeräte nicht zu überlasten.

Leistungseffizienz

In diesem Abschnitt wird beschrieben, wie wir diese Lösung unter Verwendung der Prinzipien und bewährten Verfahren des Pfeilers Leistungseffizienz konzipiert haben.

- Die Lösung verwendet DynamoDB, eine vollständig verwaltete serverlose NoSQL-Datenbank mit On-Demand-Skalierung.

- Die Lösung verwendet Amazon S3 für die Objektspeicherung und zum Hosten einer Website (durch CloudFront), um kostengünstig, skalierbar und mit einer Lebensdauer von 11 bis 9 Sekunden zu sorgen.

Kostenoptimierung

In diesem Abschnitt wird beschrieben, wie wir diese Lösung unter Verwendung der Prinzipien und bewährten Methoden des Pfeilers Kostenoptimierung konzipiert haben.

- Wo immer möglich, haben wir die Lösung so entwickelt, dass sie eine serverlose Architektur verwendet. Sie zahlen also nur für das, was Sie tatsächlich nutzen.

Nachhaltigkeit

In diesem Abschnitt wird beschrieben, wie wir diese Lösung unter Verwendung der Prinzipien und bewährten Verfahren der Säule Nachhaltigkeit konzipiert haben.

- Die modulare, komponentenbasierte Architektur der Lösung bietet die Flexibilität, Ressourcen so anzupassen, dass sie für individuelle Anwendungsfälle bereitgestellt werden.
- Die Architektur verwendet serverlose Rechenleistung und Speicherung, wodurch die Ressourcennutzung optimiert wird.
- Als cloudbasierte Lösung profitiert diese Lösung von gemeinsam genutzten Ressourcen, Netzwerken, Stromkühlung und physischen Einrichtungen.

Einzelheiten zur Architektur

In diesem Abschnitt werden die Komponenten und AWS-Services beschrieben, aus denen diese Lösung besteht, sowie die Architekturdetails dazu, wie diese Komponenten zusammenarbeiten.

AWS-Services in dieser Lösung

AWS Service	Description
Amazon API Gateway	Kern. Dieser Service stellt den REST APIs für das Deployment-Dashboard und die WebSocket API für den Anwendungsfall bereit.
AWS CloudFormation	Kern. Diese Lösung wird als CloudFormation Vorlage verteilt und CloudFormation stellt die AWS-Ressourcen für die Lösung bereit.
Amazon CloudFront	Kern. CloudFront stellt die in Amazon S3 gehosteten Webinhalte bereit.
Amazon Cognito	Kern. Dieser Dienst kümmert sich um die Benutzerverwaltung und Authentifizierung für die API.
Amazon-DynamoDB	Kern. DynamoDB speichert Bereitstellungsinformationen und Konfigurationsdetails für das Deployment-Dashboard. Es speichert den Chatverlauf und die Konversation IDs im Text-Anwendungsfall, um den Konversationsverlauf und die Abfrageunterscheidung zu ermöglichen.
AWS Lambda	Kern. Die Lösung verwendet Lambda-Funktionen für: * Unterstützung der REST- und WebSocket API-Endpunkte * Verwaltung der Kernlogik

AWS Service	Description
	jedes Anwendungsfall-Orchestrators * Implementierung benutzerdefinierter Ressource n während der Bereitstellung CloudFormation
Amazon S3	Kern. Amazon S3 hostet die statischen Webinhalte.
Amazon CloudWatch	Unterstützend. Diese Lösung veröffent licht Protokolle aus Lösungsressourcen in CloudWatch Logs und veröffentlicht Metriken in CloudWatch Metriken. Die Lösung erstellt auch ein CloudWatch Dashboard zur Anzeige dieser Daten.
AWS Systems Manager	Unterstützend. Systems Manager bietet Ressourcenüberwachung und Visualisi erung von Ressourcenoperationen und Kostendaten auf Anwendungsebene. Wird auch zum Speichern von Konfigurationsdaten im Parameter Store verwendet.
AWS WAF	Unterstützend. AWS WAF wird vor der API Gateway Gateway-Bereitstellung bereitgestellt, um diese zu schützen.
Amazon Bedrock	Optional. Die Lösung nutzt Amazon Bedrock für den Zugriff auf grundlegende oder maßgeschn eiderte Modelle, Amazon Bedrock Agents und Amazon Bedrock Knowledge Bases. Amazon Bedrock ist die empfohlene Integration, um zu verhindern, dass Ihre Daten das AWS-Netzw erk verlassen.

AWS Service	Description
Amazon Bedrock AgentCore	Optional Die Lösung nutzt Amazon Bedrock, AgentCore um MCP-Serververbindungen sowie Agent Builder- und Workflow-Anwendungsfälle auszuführen und zu unterstützen.
Amazon Elastic Container Registry (Amazon ECR)	Optional. Bei Agent Builder-Bereitstellungen speichert und verteilt ECR Agenten-Container-Images. Die Lösung verwendet den ECR Pull-Through Cache, um automatisch vorgefertigte Agenten-Images aus dem öffentlichen ECR-Repository des GAAB-Teams abzurufen.
AWS-Distribution für OpenTelemetry (ADOT)	Optional. Für Agent Builder-Bereitstellungen bietet ADOT eine automatische Instrumen tierung für die Beobachtbarkeit der Agenten und ermöglicht so eine verteilte Ablaufver folgung und strukturierte Protokollierung für Agentenoperationen.
Amazon Kendra	Optional. Im Text-Anwendungsfall können Admin-Benutzer optional entscheiden, einen Amazon Kendra Kendra-Index zu verbinden, um ihn als Wissensdatenbank für die Konversat ion mit dem LLM zu verwenden. Dies kann verwendet werden, um neue Informationen in das LLM einzufügen, sodass es diese Informati onen in seinen Antworten verwenden kann.

AWS Service	Description
Amazon SageMaker KI	Optional. Die Lösung kann in einen Amazon SageMaker AI-Inferenzendpunkt integriert werden FMs , um darauf zuzugreifen, die in Ihrem AWS-Konto und Ihrer Region gehostet werden. Sie ist eine bevorzugte Integration, um zu verhindern, dass Ihre Daten das AWS-Netzwerk verlassen. **Note** Sie müssen die Lösung in derselben Region bereitstellen, in der der Inferenzendpunkt verfügbar ist.
Amazon Virtual Private Cloud	Optional. Die Lösung bietet die Möglichkeit, Komponenten mit einer VPC-fähigen Konfiguration bereitzustellen. Bei der Bereitstellung der Lösung mit einer VPC-fähigen Konfiguration haben Sie die Möglichkeit, die Lösung eine VPC für Sie erstellen zu lassen oder eine bestehende VPC zu verwenden, die in demselben Konto und derselben Region vorhanden ist, in der die Lösung bereitgestellt wird (Bring Your Own VPC). Wenn die Lösung die VPC erstellt, erstellt sie die erforderlichen Netzwerkkomponenten, darunter Subnetze, Sicherheitsgruppen und deren Regeln, Routing-Tabellen, Netzwerk, NAT-Gateways ACLs, Internet-Gateways, VPC-Endpunkte und deren Richtlinien.

Bereitstellungs-Dashboard

Benutzerdefinierte API Gateway Gateway-Autorisierer

Unter der Oberfläche werden benutzerdefinierte Lambda-Autorisierer für API Gateway für alle API-Aufrufe (RESTful sowohl als auch WebSocket basierend) verwendet, um zu überprüfen, ob ein bestimmter Benutzer berechtigt ist, eine Aktion auf der Grundlage der Gruppe (n) auszuführen, zu der er gehört. Dieser benutzerdefinierte Autorisierer wird von einer DynamoDB-Tabelle unterstützt, die die Richtlinien für jede Gruppe enthält. Beim Aufruf einer API ruft API Gateway die benutzerdefinierte Autorisierungs-Lambda-Funktion auf, die das bereitgestellte Amazon Cognito Cognito-Zugriffstoken dekodiert, um festzustellen, zu welchen Benutzergruppen der Benutzer gehört. Die Richtlinientabelle wird dann nach dem Gruppennamen abgefragt, um die entsprechende Richtlinie für diese Gruppe zurückzugeben.

Bei jeder Bereitstellung eines neuen Anwendungsfalls wird die Admin-Richtlinie aktualisiert, sodass eine neue Anweisung gespeichert wird, die die Aktion Execute-API:Invoke für die API dieses Anwendungsfalls ermöglicht. Wenn Anwendungsfälle gelöscht werden, wird die entsprechende Anweisung aus der Richtlinie entfernt.

Für die Gruppen, die für einen einzelnen Anwendungsfall erstellt wurden, ist nur eine einzige Anweisung in der Richtlinie enthalten, sodass die Aktion Execute-API:Invoke nur für die API dieses Anwendungsfalls ausgeführt werden kann.

Aufgrund dieser Struktur kann jeder Benutzer, der zur Gruppe eines Anwendungsfalls gehört, auf die API dieses Anwendungsfalls zugreifen. Ein einzelner Benutzer kann auch manuell zu mehreren Gruppen hinzugefügt werden, sodass dieser Benutzer mehrere Anwendungsfälle verwenden kann.

> **Warning**
>
> Sie können die Richtlinien für eine bestimmte Gruppe in der Richtlinientabelle auch manuell bearbeiten, wenn Sie einer vorhandenen Benutzergruppe Zugriff auf einen neuen Anwendungsfall gewähren möchten. Die Anwendungsfallgruppe wird gelöscht, wenn der Anwendungsfall gelöscht wird (auch wenn Sie manuelle Änderungen vorgenommen haben). Gehen Sie daher vorsichtig vor, wenn Sie einen Anwendungsfall löschen.

Für den Fall, dass ein Anwendungsfallstapel eigenständig (ohne die Verwendung des Bereitstellungs-Dashboards) bereitgestellt wird, wird ein Amazon Cognito Cognito-Benutzerpool für diese Bereitstellung erstellt, der einen einzelnen Benutzer mit Zugriff auf die API dieses Anwendungsfalls

enthält. Dieser Benutzerpool gehört nur zu diesem Anwendungsfall und wird nicht von anderen eigenständigen Bereitstellungen gemeinsam genutzt.

Anwendungsfall in Textform

Streaming-Unterstützung

In einer Chat-Anwendung ist die Latenz eine wichtige Kennzahl, um eine reaktionsschnelle Benutzererfahrung zu ermöglichen. Die Möglichkeit, dass LLM-Schlussfolgerungen von Sekunden bis Minuten dauern können, stellt die Frage, wie Inhalte den Kunden am besten zur Verfügung gestellt werden können, vor Herausforderungen. Aus diesem Grund ermöglichen mehrere LLM-Anbieter das Streamen von Antworten zurück an den Anrufer. Anstatt zu warten, bis die gesamte Inferenz abgeschlossen ist, bevor eine Antwort zurückgegeben wird, kann jedes Token zurückgegeben werden, wenn es verfügbar ist.

Um die Verwendung dieser Funktion zu unterstützen, wurde der Text-Anwendungsfall so konzipiert, dass eine WebSocket API zur Unterstützung des Chat-Erlebnisses verwendet wird. Dies WebSocket wird über API Gateway bereitgestellt. Die Verwendung einer WebSocket API ermöglicht es, zu Beginn einer Chat-Sitzung eine Verbindung herzustellen und Antworten über diesen Socket zu streamen. Dadurch können Frontend-Anwendungen eine bessere Benutzererfahrung bieten.

> **Note**
>
> Selbst wenn ein Modell Streaming-Unterstützung bietet, bedeutet dies nicht unbedingt, dass die Lösung Antworten über die WebSocket API zurückstreamen kann. Die Lösung muss eine benutzerdefinierte Logik aktivieren, um Streaming für jeden Modellanbieter zu unterstützen. Wenn Streaming verfügbar ist, können Admin-Benutzer enable/disable diese Funktion zum Zeitpunkt der Bereitstellung nutzen.

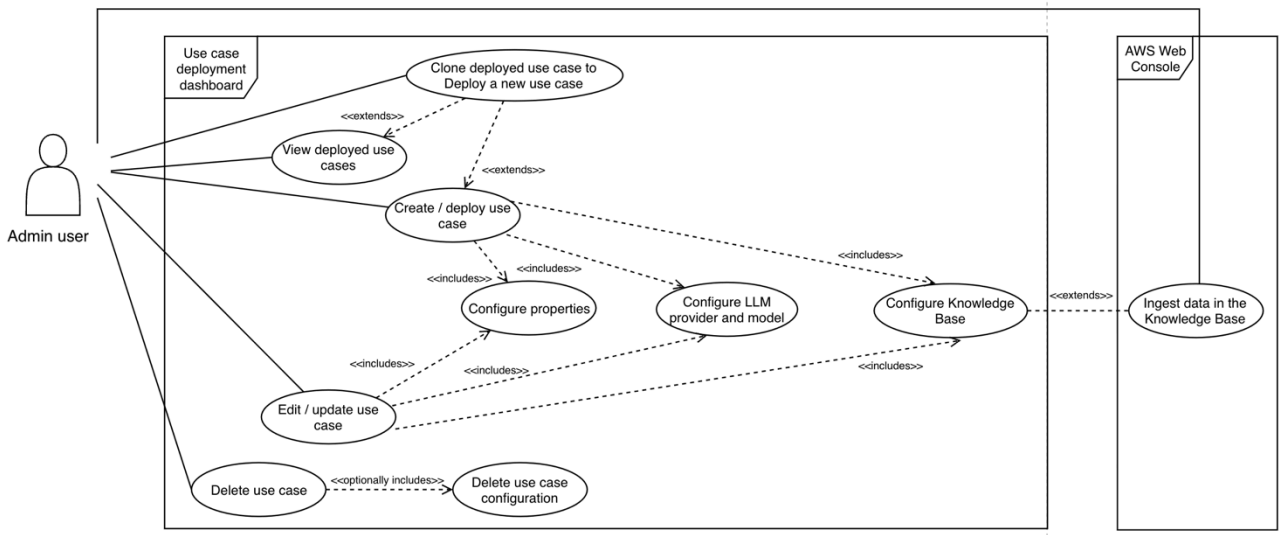
So funktioniert die Generative AI Application Builder auf AWS-Lösung

Der Admin-Benutzer ist in erster Linie mit dem Deployment-Dashboard verbunden, um neue und bestehende Anwendungsfallbereitstellungen anzuzeigen, zu erstellen und zu verwalten. Über dieses Dashboard hat der Admin-Benutzer Zugriff auf die folgenden Aktionen:

- Liste der Bereitstellungen anzeigen

- Neue Bereitstellungen erstellen
- Bestehende Bereitstellungen bearbeiten
- Klonen Sie die Konfiguration einer Bereitstellung, um eine neue Bereitstellung zu erstellen
- Löschen Sie eine Bereitstellung (heben Sie die Bereitstellung der Ressourcen durch CloudFormation Löschen auf)
- Löschen Sie die Konfigurationsdetails einer Bereitstellung dauerhaft

Zeigt ein Anwendungsfalldiagramm für den Admin-Benutzer des Deployment-Dashboards



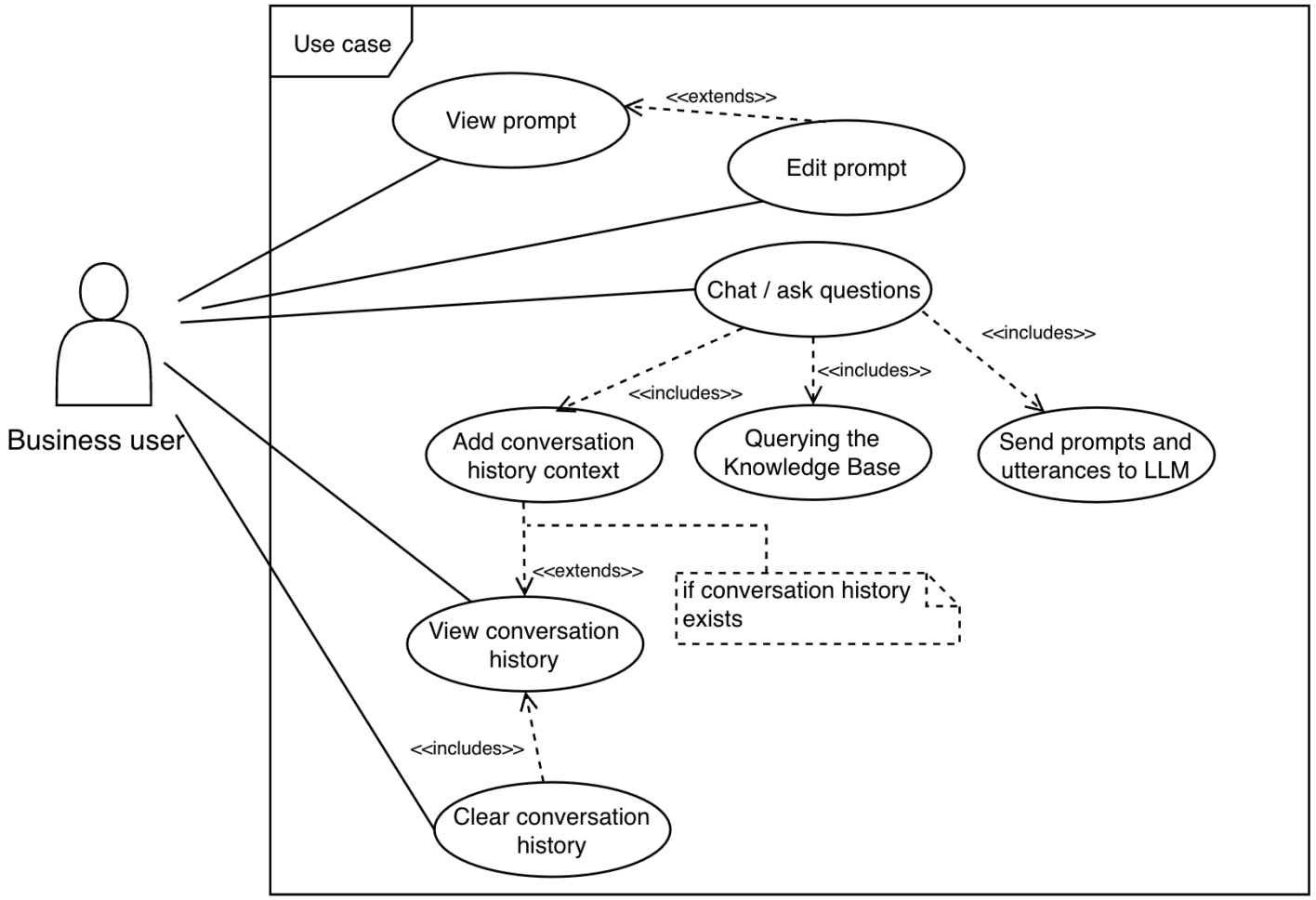
> **Note**
>
> Der Admin-Benutzer hat möglicherweise keinen direkten Zugriff auf die AWS-Konsole. In diesem Fall muss der Admin-Benutzer mit dem DevOps Benutzer zusammenarbeiten, um Aktionen wie die Aufnahme von Daten in eine Kendra-Wissensdatenbank zu unterstützen.

Für den Text-Anwendungsfall erhält der Geschäftsbenutzer Zugriff auf eine Benutzeroberfläche, über die er mit dem LLM chatten kann. Die Einzelheiten dieser Konfiguration werden durch die vom Admin-Benutzer konfigurierten Bereitstellungseinstellungen gesteuert. Im Text-Anwendungsfall hat der Geschäftsbenutzer Zugriff auf die folgenden Aktionen:

- Senden Sie Nachrichten über die Chat-Oberfläche
- Konversationsverlauf anzeigen
- Löschen Sie den Konversationsverlauf

- Aufforderung anzeigen
- Aufforderung bearbeiten

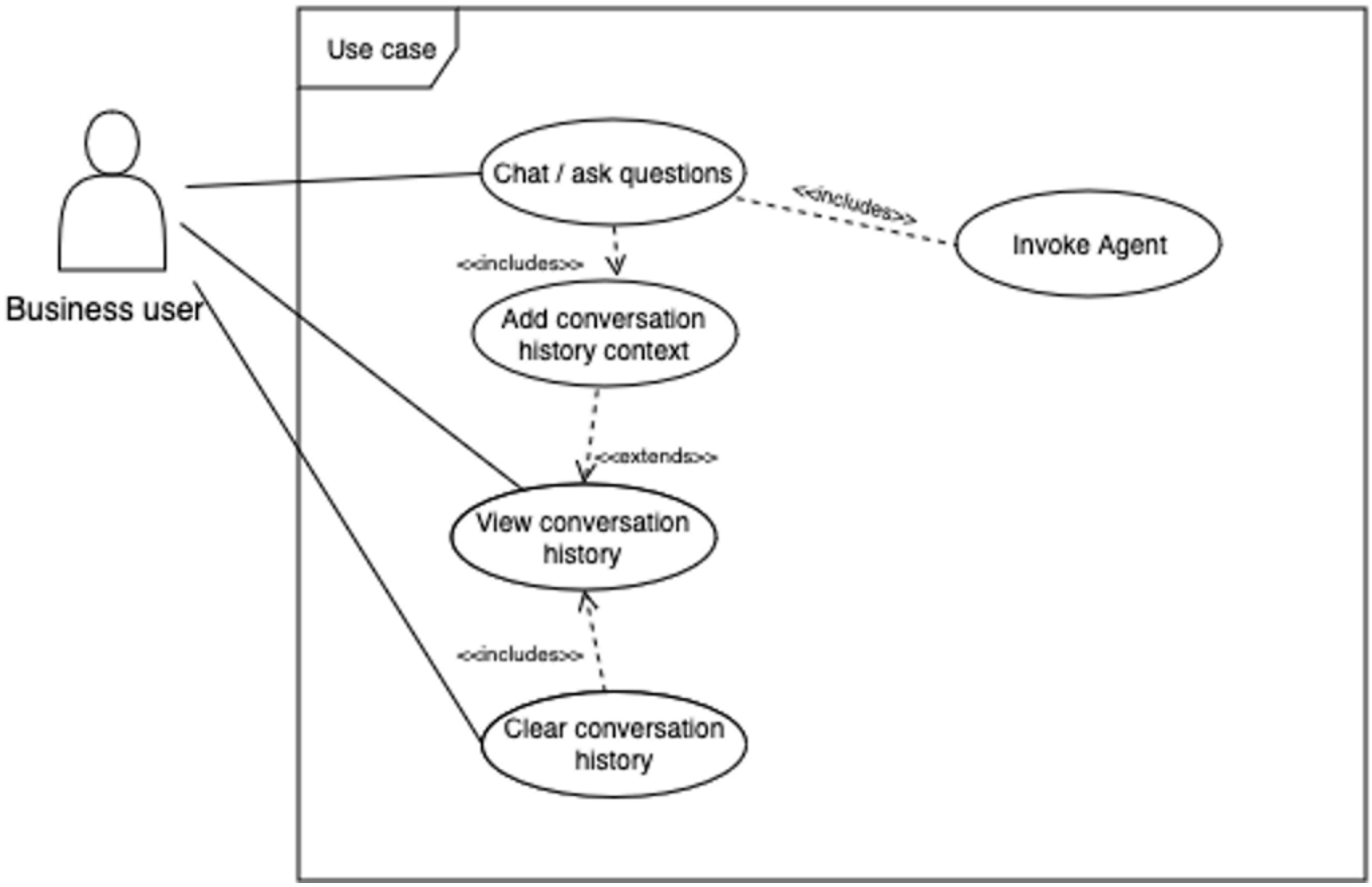
Stellt das Anwendungsfalldiagramm für den Geschäftsanwender des Text-Anwendungsfalls dar



Mit dem Bedrock Agent-Anwendungsfall kann der Geschäftsbenutzer auf eine Benutzeroberfläche zugreifen, um mit dem konfigurierten Amazon Bedrock Agent zu chatten. Der Admin-Benutzer kann diese Besonderheiten in den Bereitstellungseinstellungen konfigurieren. Im Anwendungsfall Bedrock Agent hat der Geschäftsbenutzer Zugriff auf die folgenden Aktionen:

- Senden Sie Nachrichten über die Chat-Oberfläche
- Konversationsverlauf anzeigen
- Löschen Sie den Konversationsverlauf

Stellt das Anwendungsfalldiagramm für den Geschäftsanwender des Bedrock Agent-Anwendungsfalls dar



Agent Builder

Der Agent Builder bietet eine Plattform für die Erstellung, Bereitstellung und Verwaltung produktionsbereiter KI-Agenten auf Amazon Bedrock. AgentCore In diesem Abschnitt werden die technischen Komponenten und Implementierungsdetails beschrieben.

AgentCore Integration

Agent Builder verwendet einen konfigurationsbasierten Bereitstellungsansatz mit vorgefertigten Agenten-Images, um schnelle, sichere und skalierbare Agentenbereitstellungen zu ermöglichen.

Vorgefertigte Agenten-Images

Container-Images für Agenten werden während der CI/CD Pipeline vom GAAB-Team erstellt und in einem öffentlichen ECR-Repository veröffentlicht. Jede Image-Version ist an die GAAB-Lösungsversion gebunden (z. B. v4.0.0 →:v4.0.0). gaab-strands-agent Bilder basieren auf dem Strands SDK und beinhalten:

- Laufzeitumgebung für Agenten
- MCP-Client-Integration
- Funktionen zur Speicherverwaltung
- OpenTelemetry Instrumentierung

ECR-Pull-Through-Cache

Die Lösung verwendet den ECR Pull-Through Cache, um Agenten-Images automatisch aus dem öffentlichen ECR-Repository an den privaten ECR des Kunden zu verteilen. Dieser von AWS verwaltete Service:

- Zwischenspeichert Bilder beim ersten Abruf (2-5 Minuten Verzögerung)
- Eliminiert die benutzerdefinierte Logik zum Kopieren von Bildern
- Stellt lokale Image-Verfügbarkeit für nachfolgende Bereitstellungen bereit
- Erstellt eindeutige Cache-Regeln pro Bereitstellung, um Konflikte zu vermeiden

Speicher für die Konfiguration

Agentenkonfigurationen werden zusammen mit bestehenden Anwendungsfallkonfigurationen in DynamoDB gespeichert. Jede Konfiguration umfasst:

- Vorlage für die Systemaufforderung
- Modellanbieter und Modell-ID
- Modellparameter (Temperatur, max_tokens)
- MCP-Serverreferenzen und Endpunkte
- Speichereinstellungen (Umschalten zwischen Langzeitspeicher)
- Metadaten für die Bereitstellung

Registrierung der Image-Version

Eine DynamoDB-Tabelle verfolgt die verfügbaren Agent-Image-Versionen und deren Cache URIs und ermöglicht so Versionsverwaltung und Abwärtskompatibilität.

Agentenkonfiguration

Systemaufforderungen

Systemaufforderungen definieren das Verhalten, die Persönlichkeit und die Fähigkeiten der Agenten. Admin-Benutzer können:

- Die Standardvorlage über die Agent Builder-Benutzeroberfläche bearbeiten
- Fügen Sie Anweisungen zur Verwendung des Tools und zur Formatierung von Antworten hinzu
- Sie können jederzeit auf die Standardvorlage zurückgesetzt werden

Auswahl des Modells

Agent Builder unterstützt Amazon Bedrock-Modelle in Version 4.0.0:

- Modellanbieter: Amazon Bedrock (einzige Option in v4.0.0)
- Modellauswahl: Claude, Nova und andere Bedrock-Modelle
- Modellparameter: Temperatur, max_tokens, top_p und modellspezifische Einstellungen

MCP-Serverintegration

Model Context Protocol-Server bieten Agenten Zugriff auf Unternehmenstools und Daten:

- Servererkennung über den API-Endpunkt GET /mcp
- Dynamische Konfiguration ohne Codeänderungen
- Authentifizierung und Endpunktmanagement
- Zugriff auf die Funktionen des Tools durch Agenten

Streaming und Verarbeitung

Streaming in Echtzeit

Agent Builder verwendet Server-Sent Events (SSE) von AgentCore Bridged bis hin zu WebSocket Response-Streaming in Echtzeit:

- Lambda-Funktion stellt SSE-Verbindung zu AgentCore Runtime her
- Streams werden zum API Gateway überbrückt WebSocket
- Ermöglicht die Bereitstellung von token-by-token Antworten an Kunden
- Hält die Verbindung für lang andauernde Anfragen aufrecht

Einschränkungen bei der Verarbeitung

Die Agentenverarbeitung in Version 4.0.0 ist auf das Timeout der Lambda-Ausführung beschränkt:

- Maximale Verarbeitungszeit: 15 Minuten
- Synchrones Verarbeitungsmodell
- Geeignet für Konversationsagenten und moderate Arbeitsabläufe
- Erweiterte asynchrone Unterstützung ist für Version 4.1 und höher geplant

Speicherverwaltung

Kurzzeitgedächtnis

Standardmäßig für alle Agenten aktiviert, die eine benutzerdefinierte Option verwenden MemoryHookProvider:

- Erfasst Konversationsereignisse über die Callback-Handler von Strands
- Organisiert nach ActorID und sessionId zur Kontextisolation
- Behält den Konversationskontext innerhalb von Sitzungen bei
- Automatische Integration mit AgentCore Memory

Langzeitgedächtnis

Optionale Funktion mit dem AgentCore Memory Tool von strands_tools:

- Einfaches Umschalten in der Agent Builder-Benutzeroberfläche
- Semantische Speicherstrategie mit Standardeinstellungen
- Agentengesteuerter Zugriff durch natürlichen Toolaufruf
- Speichert die gewonnenen Erkenntnisse sitzungsübergreifend
- Verwendet ConversationID als sessionId

Beobachtbarkeit

OpenTelemetry AWS-Distribution (ADOT)

Agenten werden während der Container-Erstellung automatisch instrumentiert:

- Automatische Trace-Generierung für Agentenoperationen
- Verteilte Ablaufverfolgung über Dienstgrenzen hinweg
- Strukturierte Protokollierung mit Korrelation IDs
- Integration mit der CloudWatch Transaktionssuche

Ablauf der Authentifizierung

Benutzer authentifizieren sich über Amazon Cognito mit JWT-Token, die von benutzerdefinierten Lambda-Autorisierern validiert wurden, die basierend auf Benutzergruppen IAM-Richtlinien von DynamoDB abrufen.

Workflow-Builder

Workflow Builder ermöglicht die Orchestrierung mehrerer Agenten, indem ein Supervisor-Agent erstellt wird, der mehrere Agent Builder-Agenten mithilfe des Delegierungsmusters Agents as Tools koordiniert.

Workflow-Architektur

Die wichtigsten Komponenten

- Supervisor Agent: Entrypoint-Agent, der Benutzeranfragen entgegennimmt und an spezialisierte Agenten delegiert
- Spezialisierte Agenten: Agent Builder-Anwendungsfälle, die als Tools für den Supervisor registriert sind
- Agentenregistrierung: DynamoDB-Tabelle, in der Agentenkonfigurationen und Metadaten gespeichert werden
- Orchestrierungsebene: Strands SDK-Implementierung von Agenten als Tools-Muster

Instanziierung von Agenten

Erstellung eines lokalen Agenten

Alle spezialisierten Agenten werden lokal innerhalb derselben AgentCore Runtime instanziiert:

1. Ruft Agentenkonfigurationen von DynamoDB ab
2. Erzeugt lokale Instanzen jedes Agent Builder-Agenten
3. Jeder Agent unterhält seine eigenen MCP-Serververbindungen
4. Supervisor Agent registriert spezialisierte Agenten als Tools
5. Das Strands SDK verwaltet die Auswahl und Delegierung von Agenten

Planen Sie Ihren Einsatz

In diesem Abschnitt werden die Aspekte Kosten, Sicherheit, Region und Kontingent bei der Planung Ihrer Bereitstellung beschrieben.

> **Important**
>
> Diese Lösung nutzt Amazon Bedrock als primären Service für den Zugriff auf KI-generierte Modelle. Sie müssen zunächst Zugriff auf Modelle beantragen, bevor sie in der Lösung verwendet werden können. Einzelheiten finden Sie unter Modellzugriff im Amazon Bedrock-Benutzerhandbuch.

Unterstützte AWS Regionen

> **Important**
>
> Diese Lösung verwendet optional die Services Amazon Bedrock und Amazon Kendra, die derzeit nicht in allen AWS-Regionen verfügbar sind. Sie müssen diese Lösung in einer AWS-Region starten, in der diese Services verfügbar sind. Die aktuelle Verfügbarkeit von AWS-Services nach Regionen finden Sie in der regionalen AWS-Serviceliste.

Generative AI Application Builder auf AWS wird in den folgenden AWS-Regionen unterstützt:

Name der Region	
USA Ost (Ohio)	Canada (Central)
USA Ost (Nord-Virginia)	Europa (Frankfurt)
USA West (Nordkalifornien)	Europa (Irland)
USA West (Oregon)	Europa (London)
Asien-Pazifik (Mumbai)	Europa (Milan)
Asien-Pazifik (Seoul)	Europa (Paris)

Name der Region	
Asien-Pazifik (Singapur)	Europa (Stockholm)
Asien-Pazifik (Sydney)	Middle East (Bahrain)
Asien-Pazifik (Tokio)	Südamerika (São Paulo)

> **Note**
>
> Wenn Sie in Ihren Bereitstellungen ein Foundation-Modell verwenden, auf das außerhalb von AWS zugegriffen wird, erkundigen Sie sich beim Modellanbieter, in welchen Regionen sie verfügbar APIs sind. Wenn sie nur in bestimmten Regionen verfügbar APIs sind, kann es zu Instabilität in Form von hoher Latenz oder sogar Timeouts kommen. Es ist auch wichtig, dass Sie sich bei den Rechts- und Compliance-Teams Ihres Unternehmens erkundigen, ob Daten regionale Grenzen überschreiten.

Cost (Kosten)

Mit dieser AWS-Lösung zahlen Sie nur für die Ressourcen, die Sie nutzen, und es fallen keine Mindest- oder Einrichtungsgebühren an. Benutzer zahlen für das Dashboard, mit dem Generative KI-Anwendungsfälle gestartet werden, und für alle Anwendungsfälle, die bereitgestellt werden. Die Kosten für bereitgestellte Anwendungsfälle hängen von den Konfigurationen ab. Beispielkonfigurationen:

1. Ein einfaches Bereitstellungs-Dashboard, das ungefähr 20 USD pro Monat kostet.
2. Ein einfacher, produktionsreifer Chatbot-Anwendungsfall, der mit Standardeinstellungen in den USA Ost (Nord-Virginia) bereitgestellt wird und von Amazon Bedrock ohne Zugriff auf Dokumente betrieben wird und ebenfalls etwa 200 USD pro Monat kostet.
3. Ein skaliertes System in einem Amazon VPC-Anwendungsfall, das 8.000 Abfragen pro Tag über Zehntausende von Dokumenten unterstützt, was etwa 1.500 USD pro Monat kostet. Die Kosten für den Anwendungsfall variieren je nach Konfiguration, z. B. bei Text-Anwendungsfällen mit unterschiedlichen Modellanbietern, mit oder ohne aktivierter Retrieval Augmented Generation (RAG) usw.

Beschreibung des Workloads	Geschätzte Kosten (USD/Monat)
Beispiel für die Kosten für das Bereitstellungs-Dashboard	20 USD/Monat
Beispielkosten für einen textbasierten Machbarkeitsnachweis (beinhaltet ein Bereitstellungs-Dashboard und einen Text-Anwendungsfall, ~100 Interaktionen pro Tag)	40 USD/Monat
Beispielkosten für eine hochskalierbare generative KI-Abfrage-Engine (Beinhaltet ein Bereitstellungs-Dashboard, einen Text-Anwendungsfall und einen Amazon Kendra Index für RAG, bis zu 100.000 Dokumente mit ~8.000 Abfragen pro Tag, mit aktivierter VPC)	1.500 USD/Monat
Beispielkosten für einen Machbarkeitsnachweis auf Agentenbasis (Beinhaltet ein Bereitstellungs-Dashboard, 1 Bedrock Agent-Anwendungsfall mit aktivierten Amazon Bedrock Knowledge Bases und Amazon Bedrock Guardrails, ~100 Interaktionen pro Tag)	840 USD/Monat
Beispielkosten für MCP-Server (Beinhaltet ein Bereitstellungs-Dashboard, einen MCP-Server-Anwendungsfall mit Gateway-Methode für die Lambda-Integration, ~100 Tool-Aufrufe pro Tag)	22 USD/Monat
Beispielkosten für Agent Builder	55 USD/Monat

Beschreibung des Workloads	Geschätzte Kosten (USD/Monat)
(Beinhaltet ein Bereitstellungs-Dashboard, einen Agent Builder-Anwendungsfall mit MCP-Integration und aktiviertem Langzeitspeicher, ~100 Interaktionen pro Tag)	
Beispielkosten für Workflow Builder (Beinhaltet ein Bereitstellungs-Dashboard, einen Workflow mit 3 Agent Builder-Agenten, ~100 Interaktionen pro Tag)	109 USD/Monat

> **Important**
>
> Diese Beispiele sollen Ihnen nur dabei helfen, die Kosten für Ihre spezifischen Workloads abzuschätzen. Die Nutzung verschiedener LLMs Konfigurationen oder AWS-Services kann sich auf Ihre Kosten auswirken (z. B. in serverless/on-demand billing vs. provisioned/time Rechnung gestellt). Um die Kosten zu verwalten, empfehlen wir, über AWS Cost Explorer ein Budget zu erstellen. Die Preise sind freibleibend. Vollständige Informationen finden Sie auf der Preisseite für jeden AWS-Service, der in dieser Lösung verwendet wird.

Beispielkosten für den Betrieb des Deployment-Dashboards

Die folgende Tabelle enthält die Aufschlüsselung der Kosten für ein Bereitstellungs-Dashboard mit Standardparametern und 100 aktiven Benutzern in der Region USA Ost (Nord-Virginia) für einen Monat, was etwa 20$ pro Monat kosten wird.

AWS Service	Dimensionen	Kosten [USD]
API Gateway, DynamoDB, Amazon S3 CloudFront, Lambda, Systems Manager Manager-Parameterspeicher	5.000 512 KB REST-API-Aufrufe pro Monat ohne aktiviertes Caching	1,97$

AWS Service	Dimensionen	Kosten [USD]
Amazon Cognito	100 aktive Benutzer pro Monat mit aktivierten erweiterten Sicherheitsfunktionen und ohne Benutzeranmeldung über SAML oder OIDC-Verb und	5,55$
AWS WAF	10.000 Webanfragen über eine Web-ACL und 7 definierte Regeln ohne Regelgruppen	12,60$
Gesamtkosten für das Bereitstellungs-Dashboard		20,12$

Beispielkosten für einen textbasierten Machbarkeitsnachweis

In einem Bereitstellungs-Dashboard können viele Anwendungsfälle gleichzeitig bereitgestellt werden. Die folgende Tabelle zeigt die Aufschlüsselung der Kosten eines ohne RAG bereitgestellten Anwendungsfalls für einen Geschäftsanwender, der 100 Abfragen pro Tag mit dem LLM durchführt. Abfragen werden als Textnachricht am gesendet WebSocket und die Antwort wird als Token zurückgestreamt, wobei davon ausgegangen wird, dass Streaming aktiviert ist. Bei Verwendung des Amazon Bedrock Nova Pro-Modells belaufen sich die Kosten für den Betrieb dieses Anwendungsfalls auf etwa 20 USD/Monat.

AWS Service	Dimensionen	Kosten [USD]
API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, AWS Systems Manager Parameter Store	100 Chat-Interaktionen pro Tag. Durchschnittliche Nachrichtengröße 32 KB pro Nachricht und 5 Minuten pro Verbindung.	0,61$
CloudWatch	1,5 GB CloudWatch Logs bei aktiviertem ausführlichen Modus für Experimente	7,23$

AWS Service	Dimensionen	Kosten [USD]
Amazon DynamoDB	Tabelle mit Gesprächsverlauf, 1 GB Speicher LLM-Konfigurationstabelle, 1 GB Speicher	3,05$
Zwischensumme der Kosten für den Anwendungsfall (ohne) LLMs		10,89$
Amazon Bedrock (Nova Pro)	Annahmen für 100 Interakti onen pro Tag: * Monatliche Kosten für 190.000 Eingangstoken pro Tag = 0,152 × 30$ * Monatliche Kosten für 16.000 Ausgabetokens pro Tag = 0,0512$ × 30	6,10$
Gesamtkosten der Anwendung mit Amazon Bedrock (Nova Pro)	10,89 USD (Kosten für den Anwendungsfall) + 6,10 USD (Kosten für Amazon Bedrock)	17,00$

Note

Die Kosten für Inference Calls an Services außerhalb des AWS-Netzwerks sind in diesen Schätzungen nicht enthalten. Wenn Sie keinen AWS-Modellanbieter verwenden, schlagen Sie im Preisleitfaden Ihres LLM-Anbieters nach.
Preisleitfäden für AWS-Services finden Sie unter: Amazon Bedrock-Preise und Amazon SageMaker AI-Preise.

Beispielkosten für eine hochskalierbare generative KI-Abfrage-Engine

Die folgende Tabelle enthält die Kostenaufschlüsselung eines RAG-fähigen Anwendungsfalls mit dem Nova Pro-Modell von Amazon Bedrock als LLM. Wenn eine Bedrock Knowledge Base hinzugefügt wird, kostet dieser Anwendungsfall etwa 1300 USD/Monat

AWS Service	Dimensionen	Kosten [USD]
API Gateway (WebSocket)	8000 Chat-Interaktionen pro Tag. Durchschnittliche Nachrichtengröße 32 KB pro Nachricht und 5 Minuten pro Verbindung.	38,89$
CloudFront	240.000 Anfragen pro Monat, wobei 100 GB Daten ins Internet und 1 GB Daten an den Ursprung übertragen werden	8,76$
Amazon Bedrock (Nova Pro)	Annahmen: Eingabetoken = PromptTemplate (400) + Kontext (400) + ChatHistory (1080) + Abfrage-Eingabetoken (20) = 1.900 Ausgabetokens = 160 (Durchschnitt) Bei 8.000 Transaktionen pro Tag Kosten für tägliche Eingabe-Tokens (1.900 x 8.000 = 15.200.000 Tokens x 0,0008/1000 Preis pro Token)	487,80$

AWS Service	Dimensionen	Kosten [USD]
	Kosten für tägliche Output-Token (160 x 8.000 = 1.280.000 Token x 0,0032/1000 Preis pro Token) Monatliche Kosten ((12,16$ + 4,10$) x 30)	
CloudWatch	24 Metriken mit 5 GB aufgenommenen Daten für Logs und einem Dashboard	9,72$
DynamoDB	DynamoDB-Tabelle zur Nachverfolgung des Konversationsverlaufs mit jedem Datensatz mit bis zu 1 KB Daten, 8.000 Lese- und Schreibvorgängen pro Tag	11,70$
Lambda	Containergröße: 128 MB, 512 MB kurzlebig Speicher, 2 Lambda-Funktionen, die für die Autorisierung verwendet werden Containergröße: 256 MB, 512 MB kurzlebiger Speicher, 5 Anfragen pro Sekunde mit einer durchschnittlichen Rechenzeit von 20 Sekunden	20,89$
Gesamtkosten für den Anwendungsfall		577,76 USD/Monat zzgl. Kosten für die Wissensdatenbank (siehe unten)

> **Note**
>
> Die Kosten für API-Aufrufe an Dienste außerhalb des AWS-Netzwerks sind in diesen Schätzungen nicht enthalten. Lesen Sie den Preisleitfaden Ihres LLM-Anbieters, wenn Sie Amazon Bedrock nicht verwenden.

Kosten für das Hinzufügen einer Wissensdatenbank

Die Kosten für die Wissensdatenbank variieren je nach Art der verwendeten Wissensdatenbank und (im Fall von Bedrock) nach dem von der Wissensdatenbank verwendeten unterstützenden Vektorspeicher. Die Bereitstellung und Verwaltung der Wissensdatenbanken gehört nicht zum Leistungsumfang der Lösung.

Amazon Bedrock Wissensdatenbanken

Die Lösung verwaltet oder stellt keine Ressourcen bereit, die sich auf Amazon Bedrock Knowledge Bases beziehen. Für Amazon Bedrock fallen keine Kosten für die Nutzung der Wissensdatenbank-Funktion selbst an. Ihnen wird jedoch bei jeder Anfrage die Nutzung des von Ihrem Anwendungsfall verwendeten Einbettungsmodells in Rechnung gestellt. Darüber hinaus fallen für den unterstützenden Vector Store für Ihre Wissensdatenbank (z. B. ein Index in Amazon OpenSearch Service oder eine Datenbank in Amazon Relational Database Service) Kosten an, die hier nicht angegeben oder berechnet werden können.

Für das obige Szenario mit hochskalierbarer generativer KI-Abfrage-Engine fallen für diesen Service beim Aufrufen des Amazon Bedrock Embeddings-Modells folgende Kosten an:

AWS Service	Dimensionen	Kosten [USD]
Amazon Bedrock (Amazon Titan Texteinbettungen V2)	8.000 Abfragen pro Tag mit 1.900 Eingabe-Token pro Abfrage = 15.200.000 Token = 0,30 USD pro Tag. Tageskosten x 30 Tage = 9,00 USD monatliche Kosten	9,00$

AWS Service	Dimensionen	Kosten [USD]
Beispiel für die Nutzung von Amazon OpenSearch Service (Serverless)	Serverlose Grundkonfiguration mit 4 x OpenSearch Compute Unit (OCU) (fakturierbares Minimum) = 23,04 USD pro Tag Tageskosten x 30 Tage = 691,20 USD **Note** Dies ist eine grobe Schätzung, da einige Workloads mehr erfordern werden OCUs, während für Kunden mit bereits bereitges tellten OpenSearch Ressourcen weniger Kosten anfallen werden.	691,20$
Zusätzliche Kosten insgesamt		$700,20

Amazon Kendra

Die Lösung kann einen Kendra-Index für Sie bereitstellen, oder Sie können Ihren eigenen mitbringen. Die Kosten für den Betrieb einer Konfiguration, die für die oben genannte hochskalierbare generative KI-Abfrage-Engine geeignet ist, stellen sich wie folgt dar:

AWS Service	Dimensionen	Kosten [USD]
Amazon Kendra	0-8.000 Abfragen pro Tag und bis zu 100.000 Dokumente mit	1.008,00$

AWS Service	Dimensionen	Kosten [USD]
	Amazon Kendra Enterprise Edition mit 0-50 Datenquellen	

> **Note**
>
> Sie können den Amazon Kendra Kendra-Index für mehrere Anwendungsfälle gemeinsam nutzen, dies kann jedoch die Anzahl der Abfragen pro Index erhöhen. Wenn dies nicht in die Amazon Kendra Enterprise Edition fällt, fallen zusätzliche Gebühren an.

Zusätzliche Kosten für die Aktivierung von Amazon VPC für einen Anwendungsfall

Die folgende Tabelle enthält die Aufschlüsselung der Kosten für die Aktivierung von Amazon VPC für einen Anwendungsfall, der in zwei AZs Fällen bereitgestellt wird.

AWS Service	Dimensionen	Kosten [USD]
Amazon NAT-Gateway	Annahme: 2-AZ-Bereitstellung mit einem NAT-Gateway in jeder AZ. 100 GB an über NAT Gateway verarbeiteten Daten 730 Stunden, 100 GB verarbeitete Daten pro Monat	74,70$
AWS PrivateLink (VPC-Endpunkte)	Annahmen: 2-AZ-Bereitstellung mit 1 privaten Subnetz in jeder AZ und 1 VPC-Endpunkt mit 2 elastischen Netzwerkschnittstellen ()ENIs. 6 VPC-Endpunkte, 2 ENIs pro VPC-Endpunkt, 730 Stunden	97,84$

AWS Service	Dimensionen	Kosten [USD]
	mit 1.024 GB verarbeiteter Daten in einem Monat	
Öffentliche Adresse IPv4	Annahme: 2-AZ-Bereitstellung, 1 öffentliches Subnetz in jeder AZ mit einem NAT-Gateway in jedem öffentlichen Subnetz. Jedes NAT-Gateway ist mit einem aktiven öffentlichen Gateway konfiguriert. IPv4 2 aktive öffentliche IPv4 Adressen x 730 Stunden pro Monat x 0,005$ Stundenge bühr = 7,3 USD	7,30$
Zusätzliche Kosten (für Amazon VPC)		179,93$

Auswirkungen auf die Kosten bei der Verwendung von Provisioned Throughput

Die Kosten für den bereitgestellten Durchsatz hängen von der Art des bereitgestellten Modells und Ihrem Abonnementzeitraum sowie den für den Abonnementzeitraum ausgewählten Modelleinheiten ab. Im Zusammenhang mit der Nutzung von Provisioned Throughput fallen zusätzliche Kosten an.

Weitere Informationen und die meisten up-to-date Preise finden Sie unter Bedrock Pricing.

Kosten für die Verwendung regionsübergreifender Inferenz

Bei Verwendung regionsübergreifender Inferenz fallen keine zusätzlichen Kosten für Routing oder Datenübertragung an. Sie zahlen für Modelle den gleichen Preis pro Token wie in Ihrer Quell- oder Hauptregion.

Beispielkosten für einen Machbarkeitsnachweis auf Agentenbasis

Wenn Sie Amazon Bedrock Agents verwenden, werden Ihnen Gebühren auf der Grundlage der Komponenten berechnet, aus denen der Agent besteht, wie z. B. dem Basismodell und der Wissensdatenbank (falls RAG aktiviert ist), sowie auf der Grundlage der zusätzlichen Funktionen, die Sie hinzufügen. Die folgende Tabelle zeigt die Kostenaufschlüsselung eines Bedrock Agent-Anwendungsfalls, der mit einem On-Demand-Modell Claude 3.5 Sonnet, Amazon Bedrock Knowledge Bases und Amazon Bedrock Guardrails konfiguriert wurde.

Ähnlich wie bei den Kosten für das Hinzufügen von Amazon Bedrock Knowledge Bases verwaltet oder stellt diese Lösung keine Ressourcen bereit, die sich auf Amazon Bedrock Agents beziehen. Die Lösung verursacht auch keine Kosten für die Nutzung von Amazon Bedrock Knowledge Bases, verursacht aber Kosten für:

- Verwenden des Einbettungsmodells für jede Anfrage, die an das Unternehmen gesendet wird
- Der Backing-Vector-Store für Ihre Wissensdatenbank (z. B. ein Index in Amazon OpenSearch Service oder eine Datenbank in Amazon RDS)

In der folgenden Tabelle wird von 100 Interaktionen pro Tag mit 1.900 Eingabe-Token und 160 Ausgabetokens pro Abfrage ausgegangen.

> **Note**
>
> In diesem Beispiel-Anwendungsfall für Bedrock Agent würden diese Kosten zusätzlich anfallen, wenn eine Aktionsgruppe für die Verwendung einer externen API konfiguriert wäre. Sie fallen nicht in den Rahmen der Berechnungen in dieser Tabelle.

AWS Service	Dimensionen	Kosten [USD]
API Gateway (WebSocket), Lambda CloudFront, Amazon S3, Systems Manager Manager-Parameterspeicher	100 Chat-Interaktionen pro Tag, durchschnittliche Nachrichtengröße 32 KB pro Nachricht, 5 Minuten pro Verbindung	0,61$

AWS Service	Dimensionen	Kosten [USD]
CloudWatch	1,5 GB CloudWatch Protokolle mit aktiviertem ausführlichen Modus für Experimente	7,23$
DynamoDB	LLM-Konfigurationstabelle für 1 KB Datensatzgröße und 1 GB Speicher	0,25$
Zwischensumme der Kosten (ohne) LLMs		8,09$
Anthropisches Claude 3.5 Sonett	* Tageskosten für 190.000 Eingabe-Token pro Tag (0,003/1.000 Token) = 0,57$ + Tageskosten × 30 Tage = 17,10$ * Tageskosten für 16.000 Ausgangstoken pro Tag (0,015/1.000 Token) = 0,24$ + Tageskosten × 30 Tage = 7,20$	24,30$
Amazon Bedrock (Amazon Titan Text Embeddings V2) für Amazon Bedrock Wissensdatenbanken	Tageskosten für 190.000 Eingabe-Token pro Tag (0,00002/1000 Token) = 0,004 Tageskosten × 30 Tage = 0,12$	0,12$

AWS Service	Dimensionen	Kosten [USD]
Beispiel für die Nutzung von Amazon OpenSearch Service (Serverless)	Serverlose Grundkonfiguration mit 4 × OpenSearch Compute Unit (OCU) (fakturierbares Minimum) = 23,04 USD pro Tag Tägliche Kosten × 30 Tage = 691,20 USD	691,20$
Integritätsschutz für Amazon Bedrock	190.000 Token entsprechen in etwa 760.000 (190.000 × 4) Zeichen und 3.800 Texteinheiten (760.000 Zeichen/200) Stellen Sie sich eine Leitplanke vor, die mit Inhaltsfiltern, Filtern für personenbezogene Daten (PII), Filtern vertraulicher Informationen (regulärer Ausdruck) und Wortfiltern konfiguriert ist Tägliche Kosten für den Inhaltsfilter (0,75/1000 Texteinheiten) + Kosten für den PII-Filter (0,1/1000 Texteinheiten) + Filter für vertrauliche Informationen (Regex) + Wortfilter = 2,85 USD + 0,38 USD + 0 USD Monatliche Kosten = Tageskosten × 30 Tage = 96,90 USD	96,90$

AWS Service	Dimensionen	Kosten [USD]
Gesamtantragskosten für einen Agenten, der von Anthropic Claude 3.5 Sonnet unterstützt wird	8,09 USD (Kosten für Anwendungsfälle) + 812,52 USD (andere Agentenko nfigurationen)	820,61$

> **Note**
>
> Wenn Sie keinen AWS-Modellanbieter verwenden, schlagen Sie im Preisleitfaden Ihres LLM-Anbieters nach. Preisleitfäden für AWS-Services finden Sie unter: Amazon Bedrock-Preise und Amazon SageMaker AI-Preise.

Beispielkosten für MCP-Server

MCP-Server-Anwendungsfälle ermöglichen die Bereitstellung und Verwaltung von Model Context Protocol-Servern auf Amazon AgentCore Bedrock. Die folgende Tabelle zeigt die Kostenaufschlüsselung eines MCP-Server-Anwendungsfalls, bei dem die Gateway-Methode zum Umschließen vorhandener Lambda-Funktionen verwendet wird.

Die Lösung verwaltet die Bereitstellung und Konfiguration des AgentCore Gateways. Ihnen wird Folgendes in Rechnung gestellt:

- Infrastrukturkosten (API Gateway, Lambda, DynamoDB, CloudWatch S3)
- AgentCore Gateway-Verbrauch (pro Tool-Aufruf)
- Kosten für die Ausführung von Lambda-Funktionen (für die Gateway-Methode mit Lambda-Zielen)
- Externe API-Kosten (für die Gateway-Methode mit API- oder MCP-Serverzielen, falls zutreffend)

Item	Berechnungen	Cost (Kosten)
Amazon API Gateway (REST-API)	100 Tool-Aufrufe pro Tag × 30 Tage = 3.000 Anfragen pro Monat	$0.05

Item	Berechnungen	Cost (Kosten)
AWS Lambda (Orchestrierung)	100 Aufrufe pro Tag × 30 Tage × durchschnittlich 1 Sekunde × 512 MB = 3.000 GB-Sekunden pro Monat	$0.05
Amazon DynamoDB	3.000 read/write Anfragen pro Monat + 1 GB Speicher	0,15$
Amazon CloudWatch	Standardüberwachung und Protokollierung für 3.000 Aufrufe	1,00$
Amazon S3	Konfigurationsspeicher und Protokolle (minimale Nutzung)	0,25$
Amazon Bedrock Gateway AgentCore	3.000 Tool-Aufrufe pro Monat	$0.05
Lambda-Zielfunktion	100 Aufrufe pro Tag × 30 Tage × 0,5 Sekunden × 128 MB = 1.500 GB-Sekunden pro Monat	0,25$
Monatliche Gesamtkosten	1,75$ (Infrastruktur) + 0,05$ (Gateway) AgentCore	1,80$

Note

Die Kosten variieren je nach Bereitstellungsmethode (Gateway oder Runtime), Zieltypen und Nutzungsmustern. Für Bereitstellungen mit Runtime-Methoden fallen AgentCore Runtime-Gebühren anstelle von Gateway-Gebühren an. Externe API-Kosten und Kosten für benutzerdefiniertes Container-Hosting fallen zusätzlich an.

Beispielkosten für Agent Builder

Mit Agent Builder können Sie benutzerdefinierte Agenten auf Amazon Bedrock AgentCore erstellen und bereitstellen. Die folgende Tabelle zeigt die Kostenaufschlüsselung eines Agent Builder-Anwendungsfalls, der mit Claude 3.5 Sonnet, MCP-Serverintegration und aktiviertem Langzeitspeicher konfiguriert wurde.

Die Lösung verwaltet die Bereitstellung und Konfiguration von AgentCore Runtime. Ihnen wird Folgendes in Rechnung gestellt:

- Infrastrukturkosten (API Gateway, Lambda, DynamoDB, CloudWatch S3)
- AgentCore Laufzeitverbrauch (CPU- und Arbeitsspeicherstunden basierend auf der tatsächlichen Ausführungszeit des Agenten)
- Inferenz des Basismodells (Eingabe- und Ausgabetokens)
- AgentCore Gedächtnis (kurzfristige Ereignisse und langfristiges Speichern/Abrufen)

In der folgenden Tabelle wird von 100 Interaktionen pro Tag mit 1.900 Eingabe-Tokens und 160 Ausgabetokens pro Abfrage ausgegangen, wobei die durchschnittliche Ausführungszeit des Agenten 5 Sekunden pro Interaktion beträgt.

AWS Service	Dimensionen	Kosten [USD]
API Gateway (WebSocket), Lambda CloudFront, Amazon S3, Systems Manager Manager-Parameterspeicher	100 Chat-Interaktionen pro Tag, durchschnittliche Nachrichtengröße 32 KB pro Nachricht, 5 Minuten pro Verbindung	0,61$
CloudWatch	1,5 GB CloudWatch Protokolle mit aktiviertem ausführlichen Modus für Experimente	7,23$
DynamoDB	LLM-Konfigurationstabelle für 1 KB Datensatzgröße und 1 GB Speicher	0,25$

AWS Service	Dimensionen	Kosten [USD]
Zwischensumme der Infrastrukturkosten		8,09$
Amazon Bedrock Runtime AgentCore	* CPU: 1 vCPU × 5 Sekunden × 100 Interaktionen = 125 vCPU- seconds/day = 0.140 vCPU-hours/day + Tageskosten: 0,140 × 0,0895$ = 0,013$ + Monatliche Kosten: 0,013$ × 30 = 0,38$ * Speicher: 512 MB (0,5 GB) × 5 Sekunden × 100 Interaktionen = 250 GB-seconds/day = 0.069 GB-hours/day + Tageskosten: 0,069 × 0,00945$ = 0,0007$ + Monatliche Kosten: 0,0007$ × 30 = 0,02$	0,40$
Anthropisches Claude 3.5 Sonett	* Tageskosten für 190.000 Eingabe-Token pro Tag (0,003/1.000 Token) = 0,57$ + Tageskosten × 30 Tage = 17,10$ * Tageskosten für 16.000 Ausgabetoken pro Tag (0,015/1.000 Token) = 0,24$ + Tageskosten × 30 Tage = 7,20$	24,30$

AWS Service	Dimensionen	Kosten [USD]
Amazon Bedrock Arbeitsspeicher AgentCore	* Kurzzeitgedächtnis: 100 neue events/day × 0,25 USD/1.000 Ereignisse = 0,025 USD/Tag + Monatliche Kosten: 0,025 USD × 30 USD = 0,75 USD * Langzeitspeicher (integrierte Strategie): 100 Datensätze × 0,75/1.000 $ = 0,075 USD/Monat records/month * Abruf aus dem Langzeitgedächtnis: 100 retrievals/day × 0,50 USD/1.000 Abrufe = 0,05 USD/Tag + Monatliche Kosten: 0,05 USD × 30 = 1,50 USD	2,33$
Gesamtanwendungskosten für Agent Builder mit Claude 3.5 Sonnet	8,09$ (Infrastruktur) + 0,40$ (AgentCore Laufzeit) + 24,30$ (Modell) + 2,33$ (Speicher)	35,12$

Note

AgentCore Die Runtime-Preisgestaltung richtet sich nach dem Verbrauch. Die tatsächlichen Kosten hängen ab von:

- Ausführungszeit des Agenten (CPU- und Speicherauslastung während der aktiven Verarbeitung)
- Anzahl der Interaktionen und ihre Komplexität
- Verwendung des MCP-Tools (zusätzlich CPU/memory für die Ausführung des Tools)
- Speicherkonfiguration (Kurzzeit- und Langzeitgedächtnis aktiviert)

Detaillierte AgentCore Preise finden Sie unter Amazon Bedrock Pricing.

Note

Wenn Sie MCP-Server verwenden, die externe Dienste APIs oder Dienste aufrufen, fallen diese Kosten zusätzlich an und fallen nicht in den Rahmen dieser Berechnung. Ebenso fallen bei Verwendung von AgentCore Browser- oder Code Interpreter-Tools verbrauchsabhängige Gebühren in Höhe von 0,0895 USD pro vCPU-Stunde und 0,00945 USD pro GB-Stunde an.

Beispielkosten für Workflow Builder

Workflow Builder erstellt einen Supervisor-Agenten, der mehrere Agent Builder-Agenten orchestriert. Die folgende Tabelle zeigt die Aufschlüsselung der Kosten für einen Workflow mit einem Supervisor-Agenten und 3 spezialisierten Agent Builder-Agenten, die alle mit Claude 3.5 Sonnet konfiguriert und Langzeitspeicher aktiviert sind.

Annahmen: 100 Interaktionen pro Tag, durchschnittlich 2 Agentendelegationen pro Interaktion, 5 Sekunden Ausführungszeit pro Agent.

AWS Service	Dimensionen	Kosten [USD]
API Gateway (WebSocket), Lambda CloudFront, Amazon S3, Systems Manager Manager-Parameterspeicher	100 Chat-Interaktionen pro Tag, durchschnittliche Nachrichtengröße 32 KB pro Nachricht, 5 Minuten pro Verbindung	0,61$
CloudWatch	1,5 GB CloudWatch Protokolle mit aktiviertem ausführlichen Modus für Experimente	7,23$
DynamoDB	LLM-Konfigurationstabelle für 1 KB Datensatzgröße und 1 GB Speicher	0,25$

AWS Service	Dimensionen	Kosten [USD]
Zwischensumme der Infrastrukturkosten		8,09$
Amazon Bedrock AgentCore Runtime (Supervisor Agent)	* CPU: 1 vCPU × 5 Sekunden × 100 Interaktionen = 0,140 vCPU- hours/day × 30 = $0.38 * Memory: 0.5 GB × 5 seconds × 100 interactions = 0.069 GB-hours/day × 30 = 0,02$	0,40$
Amazon Bedrock AgentCore Runtime (3 spezialisierte Agenten)	* Durchschnittlich 2 Delegationen pro Interaktion = 200 Agenten executions/day * CPU: 1 vCPU × 5 seconds × 200 = 0.278 vCPU-hours/day × 30 = $0.75 * Memory: 0.5 GB × 5 seconds × 200 = 0.139 GB-hours/day × 30 = 0,04$	0,79$
Anthropic Claude 3.5 Sonett (Beauftragter)	* Eingabe: 190.000 USD × 0,003/1.000 USD = 0,57 USD/Tag tokens/day × 30 = 17,10 USD * Ausgabe: 16 000 × 0,015 USD/1.000 USD = 0,24 USD/Tag × 30 = 7,20 USD tokens/day	24,30$
Anthropic Claude 3.5 Sonnet (Spezialisierte Agenten)	* Durchschnittlich 2 Delegationen pro Interaktion * Eingabe: 380 000 × 0,003 USD/1.000 $ = 1,14 USD/Tag tokens/day × 30 = 34,20$ * Ausgabe: 32 000 × 0,015 USD/1.000 $ = 0,48 USD/Tag × 30$ = 14,40$ tokens/day	48,60$

AWS Service	Dimensionen	Kosten [USD]
Amazon Bedrock AgentCore Memory (Supervisor Agent)	* Kurzfristig: 100 events/day × 0,25 USD/1 K × 30 = 0,75$ * Langfristige Speicherung: 100 Datensätze × 0,75 USD/1 K = 0,08$ * Langfristiger Abruf: 100 × 0,50 USD/1 K × 30 = 1,50$ retrievals/day	2,33$
Amazon Bedrock AgentCore Memory (spezialisierte Agenten)	* Kurzfristig: 200 events/day × 0,25 USD/1 K × 30 = 1,50$ * Langfristige Speicherung: 200 Datensätze × 0,75 USD/1 K = 0,15$ * Langfristiger Abruf: 200 × 0,50 USD/1 K × 30 = 3,00$ retrievals/day	4,65$
Gesamtanwendungskosten für Workflow Builder mit 3 Agenten	8,09$ (Infrastruktur) + 1,19$ (AgentCore Laufzeit) + 72,90$ (Modelle) + 6,98$ (Speicher)	89,16$

Note

- Höhere Delegationsraten erhöhen den Token-Verbrauch proportional

Detaillierte AgentCore Preise finden Sie unter Amazon Bedrock Pricing.

Sicherheit

Wenn Sie Systeme auf der AWS-Infrastruktur aufbauen, werden die Sicherheitsaufgaben zwischen Ihnen und AWS aufgeteilt. Dieses Modell der geteilten Verantwortung reduziert Ihren betrieblichen Aufwand, da AWS die Komponenten wie das Host-Betriebssystem, die Virtualisierungsebene und die physische Sicherheit der Einrichtungen, in denen die Services betrieben werden, betreibt, verwaltet und kontrolliert. Weitere Informationen zur AWS-Sicherheit finden Sie unter AWS Cloud Security.

Verwenden von Fundamentmodellen auf Amazon Bedrock

Amazon Bedrock bietet eine Sammlung von Modellen, von Amazon Nova-Modellen bis hin zu anderen führenden Foundation-Modellen (FMs). Bei Verwendung von Amazon Bedrock werden alle Modelle in der AWS-Infrastruktur gehostet. Das bedeutet, dass bei Verwendung von Amazon Bedrock als LLM-Anbieter alle Ihre Inferenzanfragen im AWS-Netzwerk verbleiben und der Netzwerkverkehr Ihre Region nicht verlässt.

> **Note**
>
> Alle über Amazon Bedrock verfügbaren Foundation-Modelle (FMs) werden direkt auf der AWS-Infrastruktur gehostet, die von AWS verwaltet wird und sich im Besitz von AWS befindet. Modellanbieter haben keinen Zugriff auf Kundendaten wie Eingabeaufforderungen und Weiterleitungen oder Amazon Bedrock-Serviceprotokolle. Weitere Informationen zur Sicherheitslage von Amazon Bedrock finden Sie unter Datenschutz in Amazon Bedrock im Amazon Bedrock-Benutzerhandbuch.

IAM-Rollen

IAM-Rollen ermöglichen es Kunden, Services und Benutzern in der AWS-Cloud detaillierte Zugriffsrichtlinien und -berechtigungen zuzuweisen. Diese Lösung erstellt IAM-Rollen, die den Lambda-Funktionen der Lösung Zugriff gewähren, um regionale Ressourcen zu erstellen.

CloudWatch Logs

Sie können den ausführlichen Modus bei der Bereitstellung eines Anwendungsfalls mithilfe der Modellauswahlseite des Deployment Dashboards unter Zusätzliche Einstellungen aktivieren. Der ausführliche Modus ermöglicht detaillierte CloudWatch Protokolle, die beim Debuggen und bei schnellen Experimenten hilfreich sein können.

> **Note**
>
> Wenn der ausführliche Modus aktiviert ist, werden auch abgerufene Dokumente aus der Wissensdatenbank (sofern RAG aktiviert ist) und Eingabeaufforderungen protokolliert, die vertrauliche Informationen enthalten können.

VPC

Die Lösung bietet zwei Optionen für die Amazon VPC-Konfiguration:

1. Lassen Sie die Lösung eine Amazon VPC für Sie erstellen.
2. Verwaltung und Bereitstellung Ihrer eigenen Amazon VPC zur Verwendung innerhalb der Lösung.

Lassen Sie die Lösung eine Amazon VPC für Sie erstellen

Wenn Sie die Option auswählen, die Lösung eine Amazon-VPC erstellen zu lassen, wird sie standardmäßig als 2-AZ-Architektur mit einem CIDR-Bereich 10.10.0.0/20 bereitgestellt. Sie haben die Möglichkeit, Amazon VPC IP Address Manager (IPAM) mit einem öffentlichen Subnetz und einem privaten Subnetz in jeder AZ zu verwenden. Die Lösung erstellt NAT-Gateways in jedem der öffentlichen Subnetze und konfiguriert Lambda-Funktionen, um die ENIsin den privaten Subnetzen zu erstellen. Darüber hinaus erstellt diese Konfiguration Routentabellen und ihre Einträge, Sicherheitsgruppen und ihre Regeln ACLs, Netzwerk- und VPC-Endpunkte (Gateway- und Schnittstellenendpunkte).

Verwaltung Ihrer eigenen Amazon VPC

Wenn Sie die Lösung mit einer Amazon VPC bereitstellen, haben Sie die Möglichkeit, eine bestehende Amazon VPC in Ihrem AWS-Konto und Ihrer Region zu verwenden. Wir empfehlen Ihnen, Ihre VPC in mindestens zwei Availability Zones verfügbar zu machen, um eine hohe Verfügbarkeit zu gewährleisten. Ihre VPC muss außerdem über die folgenden VPC-Endpunkte und die zugehörigen IAM-Richtlinien für Ihre VPC- und Routentabellenkonfigurationen verfügen.

Für ein Bereitstellungs-Dashboard Amazon VPC

1. Gateway-Endpunkt für DynamoDB.
2. Gateway-Endpunkt für S3.
3. Schnittstellen-Endpunkt für CloudWatch.
4. Schnittstellen-Endpunkt für AWS CloudFormation.

Für einen Anwendungsfall Amazon VPC

1. Gateway-Endpunkt für DynamoDB.

2. Gateway-Endpunkt für S3.
3. Schnittstellen-Endpunkt für CloudWatch.
4. Schnittstellenendpunkt für Systems Manager Parameter Store.

> **Note**
>
> Die Lösung erfordert nur`com.amazonaws.region.ssm`.

5. Schnittstellenendpunkt für Amazon Bedrock (Bedrock-Runtime, Agent-Runtime,). bedrock-agent-runtime
6. Optional: Wenn bei der Bereitstellung Amazon Kendra als Wissensdatenbank verwendet wird, ist ein Schnittstellenendpunkt für Amazon Kendra erforderlich.
7. Optional: Wenn für die Bereitstellung ein LLM unter Amazon Bedrock verwendet wird, ist ein Schnittstellenendpunkt für Amazon Bedrock erforderlich.

> **Note**
>
> Die Lösung erfordert nur. `com.amazonaws.region.bedrock-runtime`

8. Optional: Wenn bei der Bereitstellung Amazon SageMaker AI für das LLM verwendet wird, ist ein Schnittstellenendpunkt für Amazon SageMaker AI erforderlich.

> **Note**
>
> Die Lösung löscht oder ändert die VPC-Konfiguration nicht, wenn Sie die Bereitstellungsoption Bring Your Own VPC verwenden. Es werden jedoch alle gelöscht VPCs , die von der Lösung in der Option VPC für mich erstellen erstellt wurden. Aus diesem Grund müssen Sie vorsichtig sein, wenn Sie eine lösungsverwaltete VPC für mehrere Stacks/Bereitstellungen gemeinsam nutzen.
> Beispielsweise verwendet Deployment A die Option Create a VPC for me. Bereitstellung B verwendet Bring my own VPC mit der von Bereitstellung A erstellten VPC. Wenn Bereitstellung A vor Bereitstellung B gelöscht wird, funktioniert Bereitstellung B nicht mehr, da die VPC gelöscht wurde. Auch weil Deployment B die von Lambda ENIs erstellten Funktionen verwendet, kann es beim Löschen von Deployment A zu Fehlern und zur Beibehaltung von Restressourcen kommen.

Amazon CloudFront

Diese Lösung stellt eine Webkonsole bereit, die in einem Amazon S3 S3-Bucket gehostet wird. Um die Latenz zu reduzieren und die Sicherheit zu verbessern, umfasst diese Lösung eine CloudFront Distribution mit einer Ursprungszugriffsidentität. Dabei handelt es sich um einen CloudFront Benutzer, der öffentlichen Zugriff auf die Inhalte des Website-Buckets der Lösung gewährt. Weitere Informationen finden Sie unter Beschränken des Zugriffs auf Amazon S3 S3-Inhalte mithilfe einer Origin-Zugriffsidentität im Amazon CloudFront Developer Guide.

> **Note**
>
> CloudFront hat ein Soft-Quotenlimit auf Kontoebene von 20 Response-Header-Richtlinien. Diese Lösung erstellt aus Sicherheitsgründen benutzerdefinierte Richtlinien für Antwort-Header. Wenn Sie mehr als 20 Bereitstellungen von Generative AI Application Builder auf AWS oder seinen Anwendungsfällen haben, können neue Bereitstellungen fehlschlagen, weil das Kontingentlimit erreicht wird.

Um dieses Problem zu beheben, können Sie in der AWS-Servicekonsole eine Erhöhung des Kontingents für das Kontingent „Response Header Policies“ beantragen, indem Sie die folgenden Schritte ausführen:

1. Öffnen Sie die AWS-Servicequotas-Konsole.
2. Wählen Sie im Navigationsbereich AWS-Services aus.
3. Suchen Sie nach Amazon und wählen Sie es aus CloudFront.
4. Scrollen Sie zum Kontingent für Response Header Policies und wählen Sie Quotenerhöhung beantragen aus.
5. Folgen Sie den Anweisungen, um eine Erhöhung des Kontingentlimits für Ihr AWS-Konto zu beantragen.

Durch die Erhöhung des Kontingents für Response Header Policies können Sie sicherstellen, dass neue Bereitstellungen des Generative AI Application Builder auf AWS oder seiner Anwendungsfälle nicht aufgrund der Kontingentbegrenzung fehlschlagen.

Kontingente

Service Quotas, auch als Limits bezeichnet, sind die maximale Anzahl von Serviceressourcen oder -vorgängen für Ihr AWS-Konto.

Kontingente für AWS-Services in dieser Lösung

Stellen Sie sicher, dass Sie über ein ausreichendes Kontingent für jeden der in dieser Lösung implementierten Services verfügen. Weitere Informationen finden Sie unter AWS-Servicekontingente.

Verwenden Sie die folgenden Links, um zur Seite für diesen Service zu gelangen. Um die Service-Kontingente für alle AWS-Services in der Dokumentation anzuzeigen, ohne zwischen den Seiten zu wechseln, sehen Sie sich stattdessen die Informationen auf der Seite Service-Endpunkte und Kontingente in der PDF-Datei an.

Amazon AgentCore Bedrock-Kontingente

Beachten Sie bei Agent Builder-Bereitstellungen die folgenden Amazon AgentCore Bedrock-Servicekontingente:

Kontingent	USA Ost (Nord-Virginia)	Andere Regionen
Workloads für aktive Sitzungen pro Konto	1000	500
Gesamtzahl der Agenten pro Konto	1.000	1.000
Versionen pro Konto	1.000	1.000

Bereitstellen der Lösung

Diese Lösung verwendet CloudFormation AWS-Vorlagen und -Stacks, um ihre Bereitstellung zu automatisieren. Die CloudFormation Vorlage spezifiziert die in dieser Lösung enthaltenen AWS-Ressourcen und ihre Eigenschaften. Der CloudFormation Stack stellt die Ressourcen bereit, die in der Vorlage beschrieben sind.

Überblick über den Bereitstellungsprozess

Bevor Sie die Lösung auf den Markt bringen, sollten Sie sich mit den Kosten, der Architektur, der Sicherheit und anderen in diesem Handbuch erörterten Überlegungen vertraut machen.

> **Important**
>
> Wenn Sie Amazon Bedrock verwenden möchten, müssen Sie Zugriff auf Modelle beantragen, bevor sie verwendet werden können. Weitere Informationen finden Sie unter Modellzugriff im Amazon Bedrock-Benutzerhandbuch.

Zeit bis zur Bereitstellung: Ungefähr 10 Minuten

Schritt 1: Starten Sie den Deployment-Dashboard-Stack

Schritt 2: Stellen Sie einen Anwendungsfall bereit

Schritt 3: Stellen Sie mithilfe des Deployment-Dashboard-Assistenten einen Anwendungsfall bereit

Schritt 4: Konfiguration nach der Bereitstellung

Optional können Sie die Anwendungsfälle getrennt von der Lösung bereitstellen, wenn Sie die Benutzeroberfläche des Deployment-Dashboards nicht verwenden möchten oder APIs.

- Bereitstellen eines eigenständigen Text-Anwendungsfalls
- Bereitstellung eines eigenständigen Bedrock Agent-Anwendungsfalls

Sie können auch eine DynamoDB-Chat-Konfiguration angeben.

> **Important**
>
> Diese Lösung sendet Betriebsmetriken (die „Daten“) über die Verwendung dieser Lösung an AWS. Wir verwenden diese Daten, um besser zu verstehen, wie Kunden diese Lösung und die damit verbundenen Dienstleistungen und Produkte nutzen. Die Erfassung dieser Daten durch AWS unterliegt der AWS-Datenschutzrichtlinie.

CloudFormation AWS-Vorlage

Sie können die CloudFormation Vorlage für diese Lösung herunterladen, bevor Sie sie bereitstellen.

View template

genera
ai-application-builder-on-aws.template — Verwenden Sie diese Vorlage, um die Lösung und alle zugehörigen Komponenten zu starten. In der Standardkonfiguration werden die Kern- und Unterstützungslösungen bereitgestellt, die in den AWS-Services in diesem Lösungsabschnitt enthalten sind. Sie können die Vorlage jedoch an Ihre spezifischen Anforderungen anpassen.

> **Note**
>
> CloudFormation AWS-Ressourcen werden aus Konstrukten des AWS Cloud Development Kit (AWS CDK) erstellt.

Diese CloudFormation AWS-Vorlage stellt Generative AI Application Builder auf AWS in der AWS-Cloud bereit.

Schritt 1: Starten Sie den Deployment-Dashboard-Stack

Folgen Sie den step-by-step Anweisungen in diesem Abschnitt, um die Lösung zu konfigurieren und in Ihrem Konto bereitzustellen.

Zeit für die Bereitstellung: Ungefähr 10 Minuten

1. Melden Sie sich bei der AWS-Managementkonsole an und klicken
   Sie auf die Schaltfläche, um die `generative-ai-application-`

`builder-on-aws.template` CloudFormation Vorlage zu starten.

Launch solution

2. Die Vorlage wird standardmäßig in der Region USA Ost (Nord-Virginia) gestartet. Um die Lösung in einer anderen AWS-Region zu starten, verwenden Sie die Regionsauswahl in der Navigationsleiste der Konsole.

> **Note**
>
> Diese Lösung verwendet Amazon Kendra und Amazon Bedrock, die derzeit nicht in allen AWS-Regionen verfügbar sind. Wenn Sie diese Funktionen verwenden, müssen Sie diese Lösung in einer AWS-Region starten, in der diese Services verfügbar sind. Die aktuelle Verfügbarkeit nach Regionen finden Sie in der Liste der regionalen AWS-Dienste.

3. Vergewissern Sie sich auf der Seite Stack erstellen, dass sich die richtige Vorlagen-URL im Textfeld Amazon S3 S3-URL befindet, und wählen Sie Weiter.
4. Weisen Sie Ihrem Lösungsstapel auf der Seite „Stack-Details angeben“ einen Namen zu. Informationen zu Einschränkungen bei der Benennung von Zeichen finden Sie unter IAM- und STS-Grenzwerte im AWS Identity and Access Management-Benutzerhandbuch.
5. Überprüfen Sie unter Parameter die Parameter für diese Lösungsvorlage und ändern Sie sie nach Bedarf. Diese Lösung verwendet die folgenden Standardwerte.

Parameter	Standard	Description
E-Mail-Adresse des Admin-Benutzers	No	Die E-Mail-Adresse des Admin-Benutzers, der Zugriff auf das Deployment-Dashboard haben wird. Falls angegeben, werden eine Amazon Cognito Cognito-Gruppe und ein Benutzer mit Berechtigungen zur Bereitstellung und Verwaltung von Anwendungsfällen erstellt. Sie können sie auch verwendenplacehold

Parameter	Standard	Description
		`er@example.com` , um die Gruppe zu erstellen, aber nicht den Benutzer. Informationen zur Einrichtung Ihres Benutzerpools finden Sie unter Manuelle Konfiguration des Benutzerpools.
VpcEnabled	`No`	Sollte das Deployment-Dashboard innerhalb einer VPC bereitgestellt werden
CreateNewVpc	`No`	Nur verfügbar, wenn es VpcEnabledist`Yes`. Wenn der Wert ist`Yes`, erstellt der Stack die VPC und stellt die Lösung innerhalb der erstellten VPC bereit. Wenn VpcEnabledja `Yes` und CreateNewVpcistNo, müssen Sie eine bestehende VPC-Konfiguration (ExistingVpcId,, ExistingPrivateSubnetIdsExistingSecurityGroupIds, VpcAzs) angeben.
IPAMPoolId	(Optionale Eingabe)	Sie können IPAM konfigurieren und die erstellte ID als Eingabe angeben, um den IP-Adressbereich zuzuweisen, den die Bereitstellung dieses Stacks verwenden soll. Einzelheiten zu IPAM finden Sie unter So funktioniert IPAM.

Parameter	Standard	Description
Stellen Sie die Benutzeroberfläche bereit	`Yes`	Sie haben die Möglichkeit, das Deployment-Dashboard ohne die Web-Benutzeroberfläche (und die für die Webbereitstellung erforderlichen AWS-Ressourcen) bereitzustellen. In diesem Fall stellt die Lösung die gesamte Infrastruktur einschließlich der REST-API-Endpunkte bereit. Diese Option ist nützlich, um Ihre eigene Weboberfläche in das Deployment-Dashboard APIs zu integrieren.
ExistingVpcId	(Optionale Eingabe)	Nur erforderlich, wenn Sie die Lösung in einer vorhandenen VPC bereitstellen möchten, die Sie erstellt haben.
ExistingPrivateSubnetIds	(Optionale Eingabe)	Nur erforderlich, wenn Sie die Lösung in einer vorhandenen VPC bereitstellen möchten, die Sie erstellt haben. Die Lambda-Funktionen werden in diesem Subnetz bereitgestellt.

Parameter	Standard	Description
ExistingSecurityGroupIds	(Optionale Eingabe)	Nur erforderlich, wenn Sie die Lösung in einer vorhandenen VPC bereitstellen möchten, die Sie erstellt haben. Stellen Sie sicher, dass Sicherhei tsgruppen über die Berechtig ungen für eine ausgehende TCP-Verbindung verfügen.
VpcAzs	(Optionale Eingabe)	Nur erforderlich, wenn Sie die Lösung in einer vorhandenen VPC bereitstellen möchten, die Sie erstellt haben.
CognitoDomainPrefix	(Optionale Eingabe)	Nur erforderlich, wenn Sie die Lösung in einem vorhanden en Amazon Cognito Cognito-Benutzerpool bereitstellen möchten, den Sie erstellt haben. Wenn Sie keinen Wert angeben, generiert die Lösung ihn.
ExistingCognitoUserPoolId	(Optionale Eingabe)	Nur erforderlich, wenn Sie die Lösung in einem vorhanden en Amazon Cognito Cognito-Benutzerpool bereitstellen möchten, den Sie erstellt haben.

Parameter	Standard	Description
ExistingCognitoUserPoolClient	(Optionale Eingabe)	Nur erforderlich, wenn Sie die Lösung in einem vorhandenen Amazon Cognito Cognito-Benutzerpool bereitstellen möchten, den Sie erstellt haben. Wenn Sie keinen Wert angeben, erstellt die Lösung einen Benutzerpool-Client. Dieser Parameter kann nur angegeben werden, wenn Sie einen ExistingCognitoUserPoolIdWert angeben.

6. Wählen Sie Weiter aus.
7. Wählen Sie auf der Seite Configure stack options (Stack-Optionen konfigurieren) Next (Weiter) aus.
8. Überprüfen und bestätigen Sie auf der Seite Überprüfen und erstellen die Einstellungen. Markieren Sie das Kästchen, um zu bestätigen, dass die Vorlage AWS Identity and Access Management (IAM) -Ressourcen erstellt.
9. Wählen Sie Senden, um den Stack bereitzustellen.

Sie können den Status des Stacks in der CloudFormation AWS-Konsole in der Spalte Status anzeigen. Sie sollten in etwa 10 Minuten den Status CREATE_COMPLETE erhalten.

Schritt 2: Implementieren Sie einen Anwendungsfall

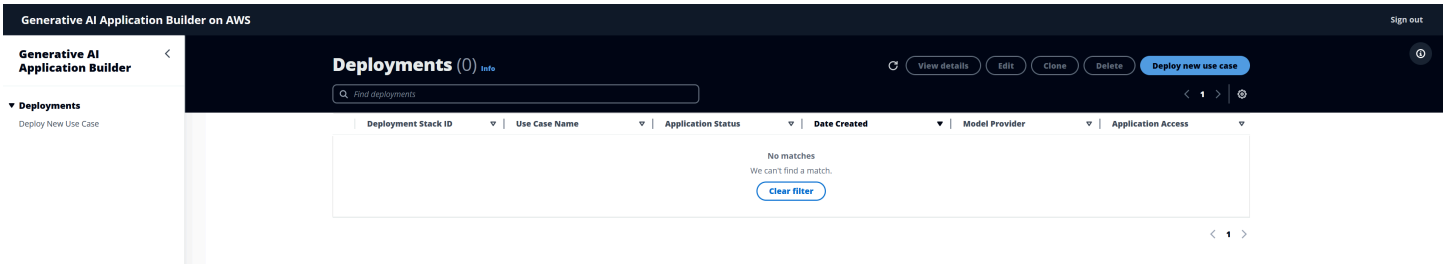
> **Important**
>
> Sobald der Stack erfolgreich bereitgestellt wurde, wird eine Anmelde-E-Mail an die konfigurierte Admin-Benutzer-E-Mail gesendet. Mit diesen Anmeldeinformationen kann sich der Admin-Benutzer im Deployment-Dashboard anmelden, um die Webanwendung zu verwenden.

> **Note**
>
> Der DevOps Benutzer mit Zugriff auf die AWS-Managementkonsole muss dem Admin-Benutzer die CloudFront URL der Deployment-Dashboard-Benutzeroberfläche mitteilen, wenn der Stack abgeschlossen ist. Die URL finden Sie auf der Registerkarte Outputs des CloudFormation Stacks.

1. Melden Sie sich als Admin-Benutzer im Deployment-Dashboard an.
2. Wählen Sie auf der Landingpage der Anwendung die Option Neuen Anwendungsfall bereitstellen aus.

   Dadurch wird der Bereitstellungsassistent gestartet, der Sie durch die Erstellung des Anwendungsfalls führt.

Zeigt die Landingpage des Deployment-Dashboards — Neuinstallation



> **Note**
>
> Wenn Sie zusätzliche Benutzer zu Ihrer Bereitstellung hinzufügen müssen, finden Sie weitere Informationen unter Verwaltung des Cognito-Benutzerpools.

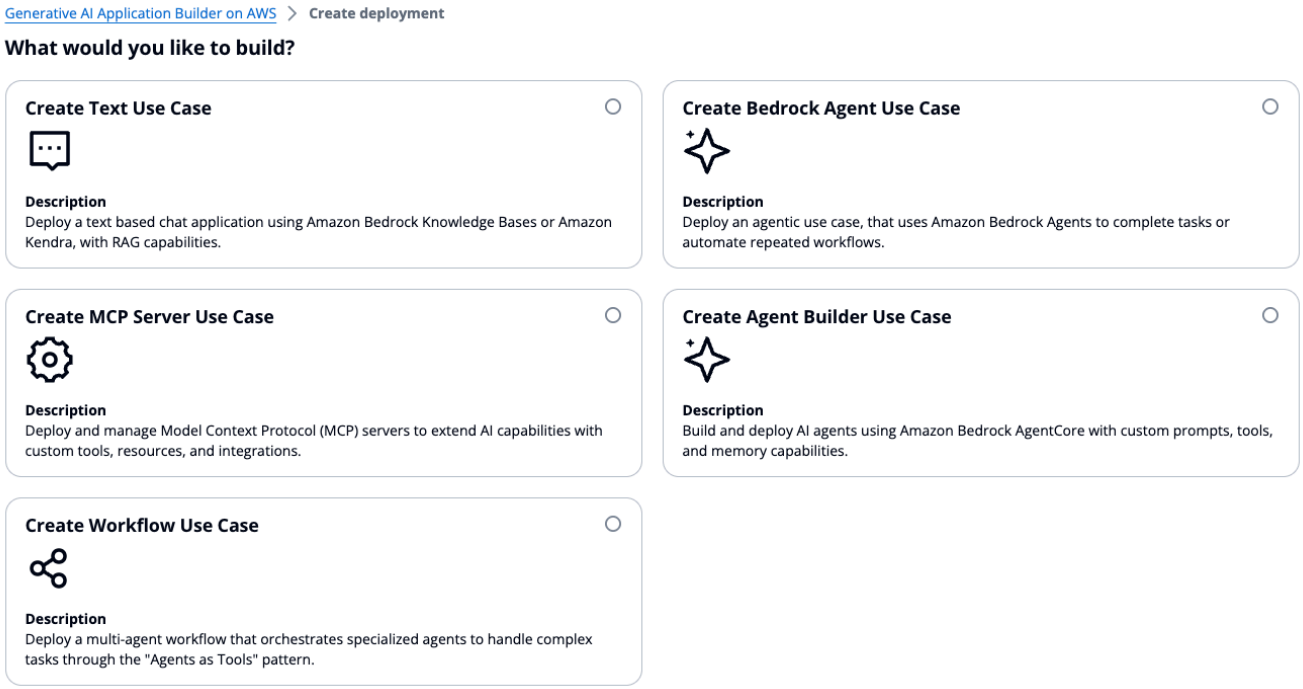
Schritt 3: Stellen Sie mithilfe des Assistenten für das Bereitstellungs-Dashboard einen Anwendungsfall bereit

Im Assistenten für das Bereitstellungs-Dashboard müssen Sie zwischen den folgenden Optionen wählen:

- Anwendungsfall Text — Stellt eine Chat-Anwendung mit optionalen RAG-Funktionen bereit

- Anwendungsfall Bedrock Agent — Nutzt Amazon Bedrock Agents, um Aufgaben zu erledigen oder sich wiederholende Workflows zu automatisieren
- MCP-Server — Stellen Sie MCP-Server mit Gateway- oder Runtime-Methoden bereit und verwalten Sie sie
- Agent Builder — Erstellen und implementieren Sie benutzerdefinierte Agenten AgentCore mit MCP-Integration und Speicherverwaltung
- Workflow Builder — Orchestrieren Sie mehrere Agent Builder-Agenten mithilfe hierarchischer Delegierung

Zeigt fünf Optionen: Text-Anwendungsfall erstellen, Bedrock Agent-Anwendungsfall erstellen, MCP-Server-Anwendungsfall erstellen, Agent Builder-Anwendungsfall erstellen oder Workflow-Anwendungsfall erstellen.



Schritt 3a: Stellen Sie einen Text-Anwendungsfall bereit

Dieser Abschnitt enthält Anweisungen zur Bereitstellung eines Text-Anwendungsfalls.

Wählen Sie einen Anwendungsfall

Wenn Sie „Text-Anwendungsfall erstellen“ wählen, öffnet die Benutzeroberfläche den Bildschirm „Anwendungsfall auswählen“. Geben Sie die folgenden Informationen ein:

- Name des Anwendungsfalls.
- Optionale E-Mail-Adresse für den Standardbenutzer des Anwendungsfalls, der dem Amazon Cognito Cognito-Benutzerpool für den Anwendungsfall hinzugefügt werden soll und dem er Berechtigungen zur Interaktion mit diesem erhält.
- Ob Sie für diesen Anwendungsfall eine Benutzeroberfläche bereitstellen möchten. Wenn Sie keine Benutzeroberfläche mit dem Anwendungsfall bereitstellen möchten, können Sie die bereitgestellten API-Endpunkte für die Verwendung mit Ihrer Anwendung verwenden.

Details zu Anwendungsfällen

Im Schritt mit den Anwendungsfalldetails können Sie zusätzliche Einstellungen für Ihre Bereitstellung konfigurieren.

Standardmäßig erstellt und konfiguriert der Text-Anwendungsfall einen Amazon Cognito Cognito-Benutzerpool für Sie, wenn die Lösung das Deployment-Dashboard bereitstellt. Die Lösung authentifiziert neue Anwendungsfälle mit einem neu erstellten Client im selben Benutzerpool. Sie können in diesem Schritt jedoch eine bestehende Benutzerpool-ID und Client-ID angeben, wenn Sie Ihren eigenen Amazon Cognito Cognito-Benutzerpool und Client für den Anwendungsfall verwenden möchten.

Important

Admin-Benutzer haben Zugriff auf alle bereitgestellten Anwendungsfälle, wenn der Amazon Cognito Cognito-Benutzerpool über den Bereitstellungsassistenten erstellt wird. Wenn Sie während der Bereitstellung Ihren eigenen Benutzerpool bereitstellen, müssen Sie sicherstellen, dass der Administrator über die Berechtigungen für den Zugriff auf die bereitgestellten Anwendungsfälle verfügt.
Sie müssen auch die Einstellungen Zulässiger Rückruf URLs und Zulässige Abmeldung URLs in Ihren App-Clients in Cognito aktualisieren. So gehen Sie vor:

1. Navigieren Sie zur Cognito-Konsole
2. Wählen Sie User Pools (Benutzerpools) aus.
3. Wählen Sie Ihren Benutzerpool.
4. Wählen Sie im linken Menü App Clients aus.
5. Wählen Sie den App-Client aus, den Sie ändern möchten.
6. Wählen Sie den Tab Anmeldeseiten.

7. Wählen Sie Bearbeiten und fügen Sie Ihre hinzu URLs.
8. Wählen Sie Änderungen speichern aus.

Wenn Sie einem Anwendungsfall weitere Benutzer hinzufügen müssen, finden Sie weitere Informationen im Abschnitt Verwaltung des Cognito-Benutzerpools.

Wählen Sie die Netzwerkkonfiguration

Mit diesem Assistentenschritt können Sie den Anwendungsfall mit einer bereits vorhandenen oder neuen Amazon Virtual Private Cloud (Amazon VPC) bereitstellen. Wenn Sie eine bereits vorhandene VPC auswählen, müssen Sie eine VPC-ID, bis zu 16 Subnetz-IDs und bis zu 5 Sicherheitsgruppen angeben, die mit dieser VPC verwendet werden IDs sollen. Wenn Sie keine bereits vorhandene VPC verwenden, werden diese Einstellungen für Sie konfiguriert.

Auswählen eines Modells

Im Schritt Modell auswählen können Sie Ihren Modellanbieter aus dem Drop-down-Menü auswählen. Es gibt zwei Optionen: Bedrock und. SageMaker

Wenn Sie sich dafür entscheiden SageMaker, können Sie in der AI-Konsole einen SageMaker KI-Modellendpunkt erstellen und das SageMaker Eingabeschema angeben, das das Modell JSONPath für die LLM-Antwort erwartet und ausgeben wird. Sie können sich den Abschnitt Amazon SageMaker AI als LLM-Anbieter verwenden und Beispiele für SageMaker KI-Nutzlasten im Repository der GitHub Lösung ansehen.

Wenn Sie Amazon Bedrock auswählen, werden Ihnen vier Optionen angeboten:

- Schnellstartmodelle — Mit einer Sammlung von Modellen mit unterschiedlichen price/performance Eigenschaften können Sie schnell loslegen. Empfohlen für die Erstellung Ihrer ersten Apps. Mit dieser Option können Sie einen Modellnamen aus der bereitgestellten Liste auswählen.
- Andere Foundation-Modelle — Greifen Sie auf die gesamte Palette von Foundation-Modellen mit unterschiedlichen Funktionen und Spezialisierungen zu. Mit dieser Option können Sie die Modell-ID für Ihr gewünschtes Bedrock On-Demand-Foundation-Modell eingeben.
- Inferenzprofile — Inferenzprofile nutzen die regionsübergreifende Inferenz von Bedrock, um den Durchsatz zu erhöhen und die Ausfallsicherheit zu verbessern, indem sie Ihre Anfragen bei Spitzenauslastung über mehrere AWS-Regionen weiterleiten. Mit dieser Option können Sie die ID des Inferenzprofils eingeben, das Sie verwenden möchten.

- Bereitgestellte Modelle — Dedizierte Durchsatzkapazität für Produktionsworkloads, die eine konsistente Leistung erfordern. Mit dieser Option können Sie den ARN des provisioned/custom Modells eingeben, das von Amazon Bedrock verwendet werden soll.

Im Schritt zur Modellauswahl können Sie auch Ihre erweiterten Modelleinstellungen auswählen. Einzelheiten zur Konfiguration von Amazon Bedrock Guardrails, zum bereitgestellten Durchsatz für Amazon Bedrock und zu zusätzlichen Modellparametern finden Sie unter Erweiterte LLM-Einstellungen.

Regionsübergreifende Inferenz

Regionsübergreifende Inferenz hilft Amazon Bedrock-Benutzern, ungeplante Datenverkehrsspitzen nahtlos zu bewältigen, indem sie Rechenleistung in verschiedenen AWS-Regionen nutzen. Um regionsübergreifende Inferenz verwenden zu können, benötigen Sie das Inferenzprofil. Ein Inferenzprofil ist eine Abstraktion über einen On-Demand-Ressourcenpool aus einer konfigurierten Gruppe von AWS-Regionen. Es kann Ihre Inferenzanfrage, die aus Ihrer Quellregion stammt, an eine andere Region weiterleiten, die in diesem Pool konfiguriert ist. Dies ermöglicht die Verteilung des Datenverkehrs auf mehrere AWS-Regionen. Dies trägt zu einem höheren Durchsatz und einer verbesserten Ausfallsicherheit in Zeiten hoher Anforderungen bei.

Inferenzprofile sind nach dem Modell und den Regionen benannt, die sie unterstützen. Sie müssen ein Inferenzprofil aus einer der Regionen aufrufen, die es enthält. Wie in der folgenden Tabelle dargestellt, `us.anthropic.claude-3-haiku-20240307-v1:0` ermöglicht die ID des Inferenzprofils beispielsweise die Verteilung des Datenverkehrs auf `us-east-1` verschiedene `us-west-2` Regionen des ausgewählten Modells. Bestimmte Modelle sind nur mit einem Inferenzprofil in einer bestimmten Region verfügbar.

Inferenzprofil	ID des Inferenzprofils	Eingeschlossene Regionen
US Anthropic Claude 3 Haiku	`us.anthropic.claude-3-haiku-20240307-v1:0`	USA Ost (Nord-Virginia) (`us-east-1`) USA West (Oregon) (`us-west-2`)

Wenn Sie eine Inferenzprofil-ID anstelle einer Modell-ID verwenden möchten, müssen Sie die entsprechende Inferenzprofil-ID identifizieren. Weitere Informationen finden Sie unter Unterstützte

Regionen und Modelle für Inferenzprofile im Amazon Bedrock-Benutzerhandbuch. In der Amazon Bedrock-Konsole stellt die Option für regionsübergreifende Inferenzen im linken Navigationsmenü diese Inferenzprofile bereit. IDs

Nachdem Sie die zu verwendende Inferenzprofil-ID identifiziert haben, können Sie diese in der Phase Modell auswählen verwenden, indem Sie die folgenden Schritte ausführen:

1. Wählen Sie Amazon Bedrock als Modellanbieter aus.
2. Wählen Sie das Optionsfeld „Inference Profiles“ aus.
3. Geben Sie Ihre Inferenzprofil-ID in das angezeigte Textfeld ein.

Weitere Informationen zu Inferenzprofilen finden Sie unter Verbessern der Resilienz mit regionsübergreifender Inferenz im Amazon Bedrock-Benutzerhandbuch.

Wählen Sie die Wissensdatenbank aus

Wenn Sie einen Anwendungsfall ohne Retrieval Augmented Generation (RAG) implementieren möchten, können Sie diesen Schritt überspringen.

Wenn Sie RAG jedoch als Teil Ihrer Bereitstellung aktivieren möchten, können Sie jetzt entweder eine vorkonfigurierte Amazon Kendra Index-ID oder eine Amazon Bedrock Knowledge Base-ID angeben. Sie können auch einen neuen Amazon Kendra Index für die Verwendung mit der Lösung erstellen. Die Lösung unterstützt derzeit Amazon Kendra und Amazon Bedrock Knowledge Bases als Wissensdatenbanken für Ihre RAG-basierte Anwendungsfallbereitstellung.

Richtlinien zur Aufnahme von Daten in die Wissensdatenbank zur Verwendung mit Ihrer RAG-basierten Bereitstellung finden Sie im Abschnitt Konfiguration einer Wissensdatenbank.

Erweiterte RAG-Konfigurationen

Mit dem Assistenten können Sie erweiterte Optionen für Ihre RAG-Implementierung auswählen, z. B. die Anzahl der Dokumente, die jedes Mal abgerufen werden sollen, wenn eine Anfrage an Ihre Wissensdatenbank gesendet wird, eine statische Textantwort des LLM, wenn keine Dokumente in der Wissensdatenbank gefunden werden, ob Sie Dokumentquellen mit Ihrer LLM-Antwort für Plausibilitätsprüfungen anzeigen möchten usw. Sie können zusätzlich auch wissensdatenbankspezifische Konfigurationen für Amazon Kendra konfigurieren, z. B. Role-based Access Control (RBAC) oder Override Search Type, wenn Sie Amazon OpenSearch Serverless mit Amazon Bedrock Knowledge Bases verwenden. Weitere Informationen zu diesen erweiterten Einstellungen finden Sie im Abschnitt Erweiterte Knowledge Base-Einstellungen.

> **Note**
>
> Ihre Wissensdatenbank muss sich in demselben Konto und derselben Region befinden wie das bereitgestellte Deployment-Dashboard und die Anwendungsfall-Stacks.

Wählen Sie Eingabeaufforderungen und Token-Limits

In diesem Schritt können Sie Ihre Aufforderung für die Verwendung mit dem LLM konfigurieren. Für Eingabeaufforderungen können Platzhalter wie`{input}`, und erforderlich sein. `{history}` `{context}` Diese Platzhalter weisen das LLM an, woher Benutzereingaben, Konversationsverlauf und aus der Wissensdatenbank abgerufene Informationen stammen sollen.

- Für Bedrock-Modellanbieter muss die Systemaufforderung bereitgestellt werden, die keine Einschränkungen für einen anderen Anwendungsfall als RAG enthält. Die Aufforderung zur Begriffsklärung für den Bedrock-Modellanbieter erfordert jedoch mindestens zwei Platzhalter - und `{input}` `{history}`
- Für die Eingabeaufforderungen SageMaker Modellanbieter, System und Disambiguierung benötigen beide mindestens zwei Platzhalter — und. `{input}` `{history}`
- Für RAG-Anwendungsfälle ist für jeden Modellanbieter zusätzlich der `{context}` Platzhalter erforderlich.

Weitere Informationen finden Sie unter Konfiguration Ihrer Eingabeaufforderungen. Bei der Auswahl der Token-Limits für Ihre Eingabeaufforderungen können Sie auch den Abschnitt Tipps zur Verwaltung von Modell-Token-Limits lesen.

Aktivieren Sie die multimodale Eingabe

In diesem Schritt können Sie multimodale Eingabefunktionen für Ihren Anwendungsfall aktivieren. Wenn diese Option aktiviert ist, können Benutzer Bilder und Dokumente zusammen mit ihren Textabfragen hochladen und senden.

Unterstützte Dateitypen und Einschränkungen:

- Bilder: Bis zu 20 Bilder pro Nachricht. Jedes Bild darf nicht größer als 3,75 MB und 8.000 Pixel hoch und breit sein. Unterstützte Formate: PNG, JPEG, GIF, Webp
- Dokumente: Bis zu 5 Dokumente pro Nachricht. Jedes Dokument darf nicht größer als 4,5 MB sein. Unterstützte Formate: pdf, csv, doc, docx, xls, xlsx, html, txt, md

So verwenden Sie multimodale Eingaben:

1. Aktivieren Sie den MultimodalEnabledParameter während der Bereitstellung des Anwendungsfalls
2. In der Chat-Oberfläche können Benutzer Dateien auf zwei Arten hochladen:
   - Klicken Sie im Chat-Eingabefeld auf die Upload-Schaltfläche oder
   - Dateien direkt in die Chat-Oberfläche ziehen und dort ablegen
3. Dateien werden auf Amazon S3 hochgeladen und vom ausgewählten Modell verarbeitet
4. Hochgeladene Dateien werden nach 48 Stunden automatisch gelöscht

Verfolgung des Dateistatus:

DevOps Benutzer können Dateimetadaten in DynamoDB überwachen, einschließlich Uploadzeit und Verarbeitungsstatus. Dateien können den folgenden Status haben:

- ausstehend — Der Datei-Upload wurde initiiert, aber noch nicht abgeschlossen. Dies ist der Anfangsstatus, wenn eine vorsignierte URL generiert wird.
- hochgeladen — Die Datei wurde erfolgreich auf S3 hochgeladen und ist bereit für die Verarbeitung durch das Modell.
- gelöscht — Die Datei wurde vom Benutzer gelöscht und sollte für die Verarbeitung nicht mehr zugänglich sein.
- ungültig — Die Überprüfung der Datei hat nicht bestanden (z. B. weil der Dateityp nicht übereinstimmt oder die Sicherheitsüberprüfung fehlgeschlagen ist).

Dateien mit dem Status „Ausstehend“, die nie hochgeladen werden, werden automatisch bereinigt, wenn ihre TTL abläuft. Nur Dateien mit dem Status „Hochgeladen“ können vom Modell verarbeitet werden.

Der multimodale S3-Bucket und die DynamoDB-Metadatentabelle sind in den Ausgaben des Deployment Dashboards mit den Schlüsseln `MultimodalDataBucketName` bzw. `MultimodalDataMetadataTable` verfügbar.

> **Note**
>
> Nicht alle Modelle unterstützen multimodale Eingaben. Stellen Sie sicher, dass Ihr ausgewähltes Modell die Bild- und Dokumentenverarbeitung unterstützt, bevor Sie diese Funktion aktivieren. In der Dokumentation Unterstützte Foundation-Modelle in Amazon

Bedrock finden Sie Informationen dazu, welches Modell Image als Eingabemodalität unterstützt.

Important

Von Benutzern hochgeladene Dateien werden in Amazon S3 mit einer Lebenszyklusrichtlinie von 48 Stunden gespeichert. Metadaten zu hochgeladenen Dateien werden in Amazon DynamoDB mit einer 24-Stunden-TTL für den Konversationsverlauf gespeichert.

Überprüfen und bereitstellen

Überprüfen Sie nach diesem Schritt die ausgewählten Einstellungen und wählen Sie „Anwendungsfall bereitstellen“. Der neue Anwendungsfall wird dann bereitgestellt und in Ihrer Bereitstellungs-Dashboard-Ansicht angezeigt, sodass Sie ihn weiter verwalten können.

Schritt 3b: Stellen Sie einen Bedrock Agent-Anwendungsfall bereit

Der Bedrock Agent-Anwendungsfall bietet einen leistungsstarken und sicheren Mechanismus zum Aufrufen von Amazon Bedrock Agents in Ihren Anwendungsfällen. Diese Funktion ermöglicht es Entwicklern, die Funktionen von KI-gestützten autonomen Agenten nahtlos zu integrieren, die mehrstufige Aufgaben über verschiedene Basismodelle, Datenquellen, Softwareanwendungen und Benutzerkonversationen hinweg orchestrieren und ausführen können, während gleichzeitig robuste Sicherheitsmaßnahmen eingehalten werden.

Voraussetzungen

Bevor Sie einen Amazon Bedrock-Agenten erstellen, stellen Sie sicher, dass Sie über Folgendes verfügen:

1. Das AWS-Konto, auf dem Generative AI Application Builder auf AWS bereitgestellt wird, mit Zugriff auf die Amazon Bedrock-Konsole.
2. Entsprechende IAM-Berechtigungen zum Erstellen und Verwalten von Amazon Bedrock Agents.

Einen Amazon Bedrock Agent erstellen

Detaillierte Anweisungen zur Erstellung eines Agenten finden Sie im Amazon Bedrock-Benutzerhandbuch unter Agenten manuell erstellen und konfigurieren. Sie können Optionen konfigurieren wie:

- Anweisungen (Eingabeaufforderungen) für Ihren Agenten
- Wissensdatenbank, die verwendet wird, um zusätzliche Informationen auf der Grundlage von Benutzereingaben nachzuschlagen
- Agentenspeicher, damit sich Agenten Informationen über mehrere Sitzungen hinweg merken können (für maximal 30 Tage)

Nachdem Sie erfolgreich einen Amazon Bedrock-Agenten erstellt haben, können Sie mit dem Assistentenablauf für Generative AI Application Builder on AWS Bedrock Agent fortfahren. Wählen Sie dazu im Bereitstellungs-Dashboard die Option Neuen Anwendungsfall bereitstellen und anschließend Bedrock Agent-Anwendungsfall erstellen aus. Folgen Sie dem Assistenten und verwenden Sie die folgenden Schritte, um den Anwendungsfall zu konfigurieren.

Wählen Sie einen Anwendungsfall

Dieser Schritt entspricht dem zuvor beschriebenen Text-Anwendungsfall.

Wählen Sie die Netzwerkkonfiguration

Dieser Schritt entspricht dem zuvor beschriebenen Text-Anwendungsfall

Wählen Sie einen Agenten aus

In diesem Schritt müssen Sie die Agenten-ID und die Alias-ID des Amazon Bedrock-Agenten angeben, den Sie erstellt haben.

Schritt 3c: Implementieren Sie einen MCP-Server-Anwendungsfall

Der Anwendungsfall MCP (Model Context Protocol) Server ermöglicht Ihnen die Bereitstellung und Verwaltung von MCP-Servern, die in KI-Modelle und -Agenten integriert werden können. MCP-Server bieten eine standardisierte Möglichkeit, Tools, Ressourcen und Funktionen für KI-Anwendungen bereitzustellen. Sie können entweder MCP-Server aus vorhandenen Lambda-Funktionen erstellen und/oder benutzerdefinierte MCP-Server mithilfe von Container-Images hosten. APIs

Voraussetzungen

Stellen Sie vor der Bereitstellung eines MCP-Server-Anwendungsfalls sicher, dass Sie über Folgendes verfügen:

1. Das AWS-Konto, auf dem Generative AI Application Builder auf AWS bereitgestellt wird.
2. Entsprechende IAM-Berechtigungen zum Erstellen und Verwalten von Amazon Bedrock-Ressourcen AgentCore .
3. Abhängig von der von Ihnen gewählten Erstellungsmethode:
   - Für die Gateway-Methode (Lambda/API/MCPServer): Lambda-Funktionen, API-Endpunkte mit ihren entsprechenden Schemadateien (JSON-Format für Lambda, OpenAPI/Smithy für APIs) oder MCP-Server-URL-Endpunkte
   - Für Runtime Method (ECR): Ein Docker-Container-Image, das an Amazon ECR übertragen wurde und Ihre MCP-Serverimplementierung enthält

Methoden zur Erstellung von MCP-Servern

Die Lösung unterstützt zwei Methoden zum Erstellen von MCP-Servern:

Aus Lambda-, API- oder MCP-Server erstellen (Gateway-Methode)

Diese Methode erstellt ein MCP-Gateway, das bestehende Lambda-Funktionen APIs, REST oder externe MCP-Server umschließt und sie als MCP-Tools zugänglich macht. Das Gateway übernimmt die Protokollübersetzung zwischen MCP und Ihren vorhandenen Diensten.

- Lambda-Ziele: Integrieren Sie bestehende Lambda-Funktionen, indem Sie die Funktion ARN und eine JSON-Schemadatei bereitstellen, die das Format der Funktion beschreibt input/output
- OpenAPI-Ziele: Integrieren Sie REST APIs mithilfe von OpenAPI-Spezifikationen (JSON- oder YAML-Format) mit Unterstützung für OAuth 2.0 oder API-Schlüsselauthentifizierung
- Smithy-Ziele: Integrieren Sie, die mithilfe von Smithy-Modelldateien (.smithy- oder .json-Format) APIs definiert wurden
- MCP-Serverziele: Stellen Sie über URL-Endpunkte eine direkte Verbindung zu externen MCP-Servern her und ermöglichen so die Integration vorhandener MCP-Server ohne Bereitstellung einer neuen Infrastruktur

Sie können mehrere Ziele (bis zu 10) innerhalb eines einzigen MCP-Gateways konfigurieren, von denen jedes ein anderes Tool oder eine andere Funktion darstellt.

Hosting über ein ECR-Image (Runtime-Methode)

Diese Methode stellt einen containerisierten MCP-Server aus einem Amazon ECR-Image bereit. Verwenden Sie diesen Ansatz, wenn Sie über eine benutzerdefinierte MCP-Serverimplementierung verfügen, die als eigenständiger Service ausgeführt werden muss.

- Geben Sie den URI des ECR-Images an (muss ein Tag enthalten, `:latest` z. B. oder) `:v1.0.0`
- Konfigurieren Sie optional Umgebungsvariablen, um die Konfiguration an Ihren Container zu übergeben
- Der Container muss das MCP-Protokoll implementieren und die erforderlichen Endpunkte verfügbar machen

Bereitstellen eines MCP-Servers

Um einen MCP-Server-Anwendungsfall bereitzustellen, wählen Sie im Bereitstellungs-Dashboard die Option Neuen Anwendungsfall bereitstellen und dann MCP-Server-Anwendungsfall erstellen aus. Folgen Sie dem Assistenten und verwenden Sie die folgenden Schritte, um den Anwendungsfall zu konfigurieren.

Wählen Sie einen Anwendungsfall

Dieser Schritt entspricht dem zuvor beschriebenen Text-Anwendungsfall.

Wählen Sie die Netzwerkkonfiguration

Derzeit ist nur der öffentliche Zugriff aktiviert und VPC wird für die Netzwerkkonfiguration nicht unterstützt.

MCP-Server erstellen

In diesem Schritt konfigurieren Sie Ihre MCP-Serverbereitstellung:

Methode zur Erstellung des MCP-Servers

Wählen Sie zwischen den beiden Erstellungsmethoden:

- Aus Lambda-, API- oder MCP-Server erstellen: Erstellen Sie ein MCP-Gateway aus vorhandenen Lambda-Funktionen, API-Spezifikationen oder externen MCP-Serverendpunkten
- Hosting über ein ECR-Image: Stellen Sie einen benutzerdefinierten MCP-Server aus einem Container-Image bereit

> **Note**
>
> Die Erstellungsmethode kann nach der Bereitstellung nicht geändert werden. Wenn Sie zwischen Methoden wechseln müssen, müssen Sie einen neuen MCP-Server-Anwendungsfall bereitstellen.

Gateway-Konfiguration (für die Lambda/API/MCP Servermethode)

Wenn Sie die Gateway-Methode ausgewählt haben, konfigurieren Sie ein oder mehrere Ziele:

1. Zielname (erforderlich): Ein benutzerfreundlicher Name zur Identifizierung dieser Zielkonfiguration
2. Zielbeschreibung (optional): Eine kurze Beschreibung dessen, was dieses Ziel tut
3. Zieltyp: Wählen Sie den zu konfigurierenden Zieltyp aus:
   - Lambda: Für AWS-Lambda-Funktionen
   - OpenAPI: Für REST APIs mit OpenAPI-Spezifikationen
   - Smithy: Für Modelldefinitionen APIs mit Smithy
   - MCP-Server: Für die direkte Verbindung zu externen MCP-Servern über URL-Endpunkte
4. Schemadatei (erforderlich): Laden Sie die Schemadatei hoch, die Ihr Ziel beschreibt:
   - Für Lambda: JSON-Schemadatei, die das input/output Format beschreibt. Einzelheiten zur Erstellung von Lambda-Tool-Schemas finden Sie unter Lambda-Toolschema im Amazon AgentCore Bedrock Developer Guide.
   - Für OpenAPI: OpenAPI-Spezifikationsdatei (JSON oder YAML). Einzelheiten zu den OpenAPI-Schemaanforderungen finden Sie unter OpenAPI-Schema im Amazon Bedrock AgentCore Developer Guide.
   - Für die Modelldatei Smithy: Smithy (.smithy oder .json). Einzelheiten zur Erstellung von Smithy-Zielen finden Sie unter Erstellen von Smithy-Zielen im Amazon Bedrock AgentCore Developer Guide.
5. Lambda-Funktion ARN (für Lambda-Ziele erforderlich): Der ARN der zu integrierenden Lambda-Funktion
6. MCP-Server-URL (erforderlich für MCP-Serverziele): Der URL-Endpunkt des externen MCP-Servers, zu dem eine Verbindung hergestellt werden soll. Die URL muss richtig codiert sein und der MCP-Server muss Toolfunktionen mit den MCP-Protokollversionen 2025-06-18 unterstützen. Weitere Informationen finden Sie unter MCP-Serverziele im Amazon Bedrock AgentCore Developer Guide.

7. Ausgehende Authentifizierung (für OpenAPI-Ziele erforderlich): Konfigurieren Sie die Authentifizierung für REST-API-Aufrufe:
   - Authentifizierungstyp: Wählen Sie OAuth 2.0 oder API-Schlüssel
   - ARN des Anbieters für ausgehende Authentifizierung: Der ARN des Anmeldeinformationsanbieters im Amazon Bedrock-Tokentresor AgentCore
   - Zusätzliche Konfigurationen: Abhängig vom Authentifizierungstyp:
     - Für OAuth 2.0: Konfigurieren Sie Bereiche und benutzerdefinierte Parameter
     - Für API-Schlüssel: Geben Sie den Speicherort (Header oder Abfrageparameter), den Parameternamen und das optionale Präfix an

Sie können mehrere Ziele (bis zu 10) hinzufügen, indem Sie Weiteres Ziel hinzufügen wählen. Jedes Ziel steht für ein separates Tool oder eine separate Funktion, die von Ihrem MCP-Server bereitgestellt wird.

ECR-Konfiguration (für die ECR-Image-Methode)

Wenn Sie die Runtime-Methode ausgewählt haben, geben Sie Folgendes an:

1. ECR-Image-URI (erforderlich): Die vollständige URI Ihres Docker-Images in Amazon ECR
   - Format: `account-id.dkr.ecr.region.amazonaws.com/repository-name:tag`
   - Das Image muss sich in derselben AWS-Region wie Ihre Bereitstellung befinden
   - Ein Tag ist erforderlich (z. B.`:latest`,`:v1.0.0`)
2. Umgebungsvariablen (optional): Konfigurieren Sie Schlüssel-Wert-Paare, die zur Laufzeit an Ihren Container übergeben werden
   - Verwenden Sie diese, um Konfigurationen, Anmeldeinformationen oder benutzerdefinierte Flags bereitzustellen
   - Sie können bis zu 10 Umgebungsvariablen hinzufügen

Überprüfen und bereitstellen

Überprüfen Sie nach der Konfiguration Ihres MCP-Servers die ausgewählten Einstellungen und wählen Sie Deploy Use Case aus. Der neue MCP-Server-Anwendungsfall wird dann bereitgestellt und zur weiteren Verwaltung in Ihrer Bereitstellungs-Dashboard-Ansicht angezeigt.

> **Note**
>
> MCP-Serverbereitstellungen erstellen Ressourcen in Amazon Bedrock AgentCore, einschließlich Gateways, Laufzeiten und Workload-Identitäten. Diese Ressourcen werden automatisch von der Lösung verwaltet und bereinigt, wenn Sie den Anwendungsfall löschen.

Schritt 3d: Stellen Sie einen Agent Builder-Anwendungsfall bereit

Mit dem Agent Builder können Sie produktionsbereite KI-Agenten auf Amazon Bedrock erstellen, konfigurieren und bereitstellen. AgentCore Diese Funktion bietet die vollständige Kontrolle über das Verhalten der Agenten durch Systemaufforderungen, Modellauswahl, MCP-Serverintegration und Speicherverwaltung.

Der Bereitstellungsprozess ist in erster Linie derselbe wie bei einem Text-Anwendungsfall, mit einigen nennenswerten Unterschieden.

Wählen Sie einen Anwendungsfall

Dieser Schritt entspricht dem zuvor beschriebenen Text-Anwendungsfall.

Details zu Anwendungsfällen

Dieser Schritt entspricht dem zuvor beschriebenen Text-Anwendungsfall.

Agent konfigurieren

In diesem Schritt konfigurieren Sie die wichtigsten Agenteneinstellungen, einschließlich der Systemaufforderung, der verfügbaren servers/Strands MCP-Tools und des Speichers.

Systemaufforderung

Die Systemaufforderung definiert das Verhalten, die Persönlichkeit und die Fähigkeiten des Agenten. Sie können:

- Bearbeiten Sie die Standardvorlage für Systemaufforderungen
- Verwenden Sie die Schaltfläche Auf Standard zurücksetzen, um die ursprüngliche Vorlage wiederherzustellen
- Fügen Sie Anweisungen zur Verwendung des Tools und zur Formatierung von Antworten bei

MCP-Serverintegration (optional)

Konfigurieren Sie Model Context Protocol-Server, um Ihrem Agenten Zugriff auf Unternehmenstools und -daten zu gewähren:

1. Wählen Sie in der Dropdownliste einen der verfügbaren MCP-Server aus
2. Sehen Sie sich die verfügbaren, sofort einsatzbereiten Tools an, auf die der Agent zugreifen kann

> **Note**
>
> MCP-Server müssen vor der Bereitstellung konfiguriert und zugänglich sein. Anweisungen zur Servereinrichtung finden Sie in der MCP-Dokumentation.

Speicherkonfiguration

Konfigurieren Sie, wie der Agent Kontext und Wissen beibehält:

- Kurzzeitgedächtnis: Standardmäßig für alle Agenten aktiviert. Behält den Konversationskontext innerhalb der Sitzungen bei.
- Langzeitgedächtnis: Aktiviert diese Option, um die Extraktion und Speicherung von Erkenntnissen sitzungsübergreifend zu aktivieren. Verwendet AgentCore Speicher mit semantischer Speicherstrategie.

Überprüfen und bereitstellen

Überprüfen Sie nach diesem Schritt die ausgewählten Einstellungen und wählen Sie Anwendungsfall bereitstellen aus. Die Installation von Agent Builder ist in der Regel in 10 bis 15 Minuten abgeschlossen. Der neue Anwendungsfall wird dann in der Deployment-Dashboard-Ansicht angezeigt, sodass Sie ihn weiter verwalten können.

Schritt 3e: Implementieren Sie einen Workflow-Anwendungsfall

Mit dem Workflow Builder können Sie Supervisor-Agenten erstellen, die mehrere Agent Builder-Agenten mithilfe des Delegierungsmusters Agents as Tools orchestrieren. Mit dieser Funktion können Sie komplexe Workflows mit mehreren Agenten erstellen, indem Sie bestehende Agent Builder-Bereitstellungen wiederverwenden.

Der Bereitstellungsprozess folgt einem ähnlichen Muster wie Agent Builder, mit zusätzlichen Schritten für die Agentensuche und -auswahl.

Wählen Sie einen Anwendungsfall

Dieser Schritt entspricht dem zuvor beschriebenen Text-Anwendungsfall.

Details zu Anwendungsfällen

Dieser Schritt entspricht dem zuvor beschriebenen Text-Anwendungsfall.

Supervisor Agent konfigurieren

In diesem Schritt konfigurieren Sie den Supervisor-Agenten, der die spezialisierten Agent Builder-Agenten koordiniert.

Systemaufforderung

Die Systemaufforderung definiert, wie der Supervisor Agent Arbeit an spezialisierte Agenten delegiert. Sie können:

- Bearbeiten Sie die Standardvorlage für Systemaufforderungen
- Fügen Sie Anweisungen für die Auswahl und Delegierung von Agenten hinzu
- Definieren Sie, wie die Ergebnisse mehrerer Agenten zusammengefasst werden
- Verwenden Sie die Schaltfläche Auf Standard zurücksetzen, um die ursprüngliche Vorlage wiederherzustellen

> **Note**
>
> In der Systemaufforderung sollte klar beschrieben werden, wann und wie die einzelnen Spezialagenten eingesetzt werden. Agentenbeschreibungen sind für eine ordnungsgemäße Delegierung von entscheidender Bedeutung.

Auswahl des Modells

Wählen Sie das Basismodell für den Supervisor-Agenten aus. Der Supervisor-Agent verwendet dieses Modell für:

- Benutzeranfragen verstehen

- Wählen Sie geeignete spezialisierte Agenten aus
- Koordinieren Sie die Ausführung der Agenten
- Antworten aggregieren und formatieren

Wählen Sie spezialisierte Agenten aus

In diesem Schritt wählen Sie aus, an welche Agent Builder-Agenten der Supervisor Arbeit delegieren kann.

Agenten hinzufügen

1. Klicken Sie auf Agent hinzufügen, um das Dialogfeld zur Agentenauswahl zu öffnen
2. Wählen Sie einen oder mehrere Agent Builder-Agenten aus der Liste aus
3. Lesen Sie die Agentenbeschreibungen, die dem Supervisor zur Verfügung gestellt werden
4. Bestätigen Sie die Auswahl

> **Note**
>
> - Für Workflows ist mindestens ein Agent Builder-Anwendungsfall als spezialisierter Agent erforderlich
> - Alle spezialisierten Agenten müssen erfolgreich bereitgestellt werden, bevor der Workflow erstellt werden kann

Überprüfen und bereitstellen

Überprüfen Sie die Workflow-Konfiguration, einschließlich:

- Aufforderung und Modell des Supervisor-Agent-Systems
- Liste der spezialisierten Agenten
- Speicher-Einstellungen

Wählen Sie „Anwendungsfall bereitstellen“. Die Workflow-Bereitstellung ist in der Regel in 15 bis 20 Minuten abgeschlossen. Der neue Workflow wird in Ihrer Bereitstellungs-Dashboard-Ansicht angezeigt, sodass Sie ihn weiter verwalten können.

Schritt 4: Konfiguration nach der Bereitstellung

Dieser Abschnitt enthält Empfehlungen für die Konfiguration der Lösung nach der Bereitstellung.

Versionierung von Amazon S3 S3-Buckets, Lebenszyklusrichtlinien und regionsübergreifende Replikation

Diese Lösung erzwingt keine Lebenszykluskonfigurationen für die von ihr erstellten Buckets. Wir empfehlen Folgendes:

- Festlegung von Lebenszykluskonfigurationen für Produktionsbereitstellungen. Einzelheiten finden Sie unter Einstellung der Lebenszykluskonfiguration für einen Bucket im Amazon Simple Storage Service-Benutzerhandbuch.
- Aktivierung der Versionierung und regionsübergreifenden Replikation für Amazon S3 S3-Buckets auf der Grundlage des Anwendungsfalls, für den die Lösung bereitgestellt wird.

Amazon DynamoDB-Backups

Diese Lösung verwendet DynamoDB für verschiedene Zwecke (siehe AWS-Services in dieser Lösung). Die Lösung aktiviert keine Backups für die von ihr erstellten Tabellen. Wir empfehlen, eine Sicherungskopie dieser Funktion für Produktionsbereitstellungen zu erstellen. Weitere Informationen finden Sie unter Sichern einer DynamoDB-Tabelle und Verwenden von AWS Backup for DynamoDB.

CloudWatch Amazon-Dashboard und Alarme

Die Lösung stellt ein benutzerdefiniertes Dashboard bereit, CloudWatch um Diagramme aus benutzerdefinierten veröffentlichten Metriken und AWS-Servicemetriken zu rendern. Wir empfehlen, CloudWatch Alarme zu erstellen und Benachrichtigungen hinzuzufügen, die auf dem Anwendungsfall basieren, für den die Lösung bereitgestellt wird.

CloudWatch Amazon-Protokolle

Lambda-Protokolle sind so konfiguriert, dass sie niemals ablaufen, und API Gateway Gateway-Protokolle sind mit einem Ablauf von 10 Jahren konfiguriert. Sie können den Ablauf der jeweiligen Protokollgruppen aktualisieren, um ihn an die Aufbewahrungsrichtlinie Ihres Unternehmens anzupassen.

Benutzerdefinierte Webdomänen mit TLS v1.2 oder höheren Zertifikaten

Die Lösung stellt eine Webbenutzeroberfläche und ein Edge-optimiertes API Gateway mithilfe von CloudFront bereit. CloudFrontDie Domain erzwingt keine Zertifikate mit TLS v1.2 oder höher. Wir empfehlen, eine benutzerdefinierte Domain mit Amazon Route 53 zu erstellen, ein Zertifikat mit AWS Certificate Manager zu erstellen oder ein vorhandenes Zertifikat zu verwenden, falls Ihre Organisation über eines verfügt.

Weitere Informationen finden Sie im Amazon Route 53 Developer Guide und Choosing a minimum TLS version for a custom domain in API Gateway.

Skalierung mit Amazon Kendra

Diese Lösung bietet die Möglichkeit, Amazon Kendra zu verwenden, um eine NLP-gestützte intelligente Suche in den aufgenommenen Dokumenten durchzuführen. Sie können die Kapazität von Amazon Kendra mithilfe der folgenden CloudFormation Parameter für größere Workloads erhöhen:

Parameter	Standard	Description
Zusätzliche Abfragekapazität von Amazon Kendra	`0`	Die Menge an zusätzlicher Abfragekapazität für einen Index und eine GetQuerySuggestionsKapazität. Eine zusätzliche Kapazitätseinheit für einen Index ermöglicht ungefähr 8.000 Abfragen pro Tag.
Zusätzliche Speicherkapazität von Amazon Kendra	`0`	Die Menge der zusätzlic hen Speicherkapazität für einen Index. Eine einzelne Kapazitätseinheit bietet 30 GB Speicherplatz oder 100.000 Dokumente, je nachdem, was zuerst erreicht wird.
Amazon Kendra Ausgabe	`Developer`	Amazon Kendra bietet Developer und Enterprise

Parameter	Standard	Description
		Editions zur Erstellung von Indizes. Weitere Informationen zu den Unterschieden zwischen den Amazon Kendra Editionen finden Sie unter Amazon Kendra — Preise.

Um die Werte dieser CloudFormation Parameter zu ändern, wählen Sie die entsprechenden Werte zum Zeitpunkt der Stack-Bereitstellung aus. Weitere Informationen zu Abfrage- und Speicherkapazitätseinheiten finden Sie unter Kapazität anpassen.

> **Note**
>
> Wenn der Text-Anwendungsfall nicht mit aktiviertem RAG bereitgestellt wird, wird Amazon Kendra Kendra-Index verwendet oder erstellt.

SSO mithilfe des Idp-Verbunds einrichten

Diese Lösung ermöglicht die Integration mit externen Identitätsanbietern, die einen SAML- oder OIDC-basierten Identitätsverbund unterstützen. Wenn die Lösung bereitgestellt wird, erstellt sie einen Amazon Cognito Cognito-Benutzerpool und eine individuelle App-Client-Integration für das Deployment-Dashboard und einzelne Anwendungsfälle. Folgen Sie basierend auf dem externen Idp den Schritten im Abschnitt Konfiguration von Identitätsanbietern für Ihren Benutzerpool im Amazon Cognito Developer Guide und wählen Sie die App-Client-Integration für das Bereitstellungs-Dashboard oder den Anwendungsfall aus, mit dem Sie SSO einrichten möchten.

Um die Benutzergruppeninformationen an Wissensdatenbanken oder Vektorspeicher in einer RAG-basierten Architektur weiterzugeben, müssen Sie Benutzergruppen aus dem externen Idp den Amazon Cognito Cognito-Benutzergruppen zuordnen. Die Lösung bietet einen ersten Lambda-Funktionstrigger für das Gerüst, der der Phase vor der Token-Generierung zugeordnet werden kann. Die Lambda-Funktion hat die Datei group_mapping.json, die aktualisiert werden muss, um die Gruppenzuordnungen bereitzustellen. Informationen zu Lambda-Triggern, die von Amazon Cognito unterstützt werden, finden Sie unter Benutzerpool-Workflows mit Lambda-Triggern anpassen.

Manuelle Konfiguration des Benutzerpools

Wenn Sie während der Bereitstellung keine Administrator- oder Standardbenutzer-E-Mail weitergeben möchten, müssen Sie die entsprechenden Benutzergruppen in Amazon Cognito manuell erstellen, um die richtigen Berechtigungen sicherzustellen:

1. Erstellen Sie für das Deployment-Dashboard eine Gruppe mit dem Namen `Admin` in Ihrem Cognito-Benutzerpool.
2. Erstellen Sie für jeden Anwendungsfall eine Gruppe mit dem Namen `${UseCaseName}-Users` in Ihrem Cognito-Benutzerpool, wo `${UseCaseName}` sich der Name Ihres bereitgestellten Anwendungsfalls befindet.

Diese Gruppen sind erforderlich, damit der Autorisierungsmechanismus ordnungsgemäß funktioniert. Alle Benutzer, denen Sie Zugriff gewähren möchten, müssen den entsprechenden Gruppen hinzugefügt werden.

Wenn übergeben `placeholder@example.com` wird, wird die Cognito-Gruppe erstellt, aber Sie müssen trotzdem die zugehörigen Benutzer erstellen und sie der Gruppe zuweisen.

Anmeldebildschirm anpassen

Diese Lösung verwendet die von Amazon Cognito gehostete Benutzeroberfläche zum Rendern der Anmeldeseite. Informationen zum Anpassen der integrierten Anmeldeseite finden Sie unter Anpassen der integrierten Anmelde- und Registrierungswebseiten im Amazon Cognito Developer Guide.

Zusätzliche Sicherheitsüberlegungen

Lesen Sie sich je nach Anwendungsfall, für den Sie die Lösung einsetzen, die folgenden Sicherheitsempfehlungen durch:

- Vom Kunden verwaltete AWS-KMS-Verschlüsselungsschlüssel — Die Lösung verwendet standardmäßig von AWS verwaltete AWS-KMS-Schlüssel, da diese ohne zusätzliche Kosten erhältlich sind. Überprüfen Sie Ihren Anwendungsfall, um festzustellen, ob Sie die Lösung aktualisieren sollten, um vom Kunden verwaltete AWS-KMS-Schlüssel zu verwenden.
- Drosselungsregeln für API-Gateways — Die Lösung wird mit Standard-Drosselungsregeln auf API Gateway bereitgestellt. Basierend auf Ihrem Anwendungsfall und dem erwarteten Transaktionsvolumen empfehlen wir Ihnen, die Drosselung für zu konfigurieren. APIs Einzelheiten

finden Sie unter Drosselung von API-Anfragen für besseren Durchsatz im Amazon API Gateway Developer Guide.

- Aktivierung von AWS CloudTrail — Als empfohlene Sicherheitsmaßnahme sollten Sie die Aktivierung von AWS CloudTrail in dem AWS-Konto in Betracht ziehen, in dem die Lösung für die Protokollierung von API-Aufrufen im AWS-Konto bereitgestellt wird. Einzelheiten finden Sie im CloudTrail AWS-Benutzerhandbuch.
- Drift-Erkennung — Wir empfehlen, die Drift-Erkennung auf CloudFormation Stacks zu konfigurieren, um unbeabsichtigte oder böswillige Änderungen am bereitgestellten Lösungsstapel zu erkennen und darüber informiert zu werden. Einzelheiten finden Sie unter Implementieren eines Alarms zur automatischen Erkennung von Abweichungen in CloudFormation AWS-Stacks.
- Cognito JSON Web Tokens (JWTs) — Die Lösung verwendet von Amazon Cognito ausgestellte Token JWTs zur Authentifizierung bei den REST-API-Endpunkten. Wir haben die Lösung mit einem Ablauf von fünf Minuten für ID-Token und Zugriffstoken konfiguriert. Wenn sich ein Benutzer abmeldet, wird ihm die Möglichkeit, neue Token zu generieren, entzogen (das Aktualisierungstoken wird gesperrt). Bis zum Ablauf des aktuellen Tokens werden jedoch alle Anfragen an den API-Endpunkt erfolgreich authentifiziert, da sie über ein gültiges Token verfügen. Überprüfen Sie die Sicherheitsüberlegungen für Ihren Anwendungsfall und passen Sie die Gültigkeitsdauer des Tokens an.

Anpassen der Lebenszyklusrichtlinien:

Überprüfen Sie bei Produktionsbereitstellungen die Lebenszyklusrichtlinien und passen Sie sie an Ihre Aufbewahrungsanforderungen an. Weitere Informationen finden Sie unter Einstellung der Lebenszykluskonfiguration für einen Bucket im Amazon Simple Storage Service-Benutzerhandbuch.

Multimodaler Dateispeicher und Lebenszyklus

Wenn Sie für Ihren Anwendungsfall multimodale Eingabefunktionen MultimodalEnabled(auf`Yes`) aktiviert haben, erstellt die Lösung einen Amazon S3 S3-Bucket zum Speichern hochgeladener Dateien und eine DynamoDB-Tabelle zur Nachverfolgung von Dateimetadaten.

Standard-Lebenszyklusrichtlinien:

- S3-Dateien: Automatisch nach 48 Stunden gelöscht
- DynamoDB-Metadaten: Datensätze laufen nach 24 Stunden ab (Konversationsverlauf TTL)

Überlegungen zur Sicherheit:

- Dateien werden nach Anwendungsfall-ID, Benutzer-ID, Konversations-ID und Nachrichten-ID partitioniert und stattdessen wird eine Datei mit einem UUID-Namen gespeichert. Die Zuordnung der UUID zu Dateinamen ist in der DynamoDB-Metadatentabelle verfügbar.
- Benutzer können nur auf Dateien zugreifen, die sie im Rahmen ihrer eigenen Konversationen hochgeladen haben
- Die Überprüfung des Dateityps erfolgt mithilfe der Erkennung magischer Zahlen
- Wir empfehlen, Amazon GuardDuty Malware Protection for S3 zu aktivieren, um hochgeladene Dateien auf schädliche Inhalte zu scannen.

Bereitstellung eines eigenständigen Text-Anwendungsfalls

Folgen Sie den step-by-step Anweisungen in diesem Abschnitt, um die Lösung zu konfigurieren und in Ihrem Konto bereitzustellen.

Zeit bis zur Bereitstellung: Ungefähr 10-30 Minuten

1. Melden Sie sich bei der AWS-Managementkonsole an und klicken Sie auf die Schaltfläche, um die CloudFront Vorlage zu starten, die Sie bereitstellen möchten.

BedrockChat.vorlage	Launch solution
SageMakerChat. Vorlage	Launch solution

2. Die Vorlage wird standardmäßig in der Region USA Ost (Nord-Virginia) gestartet. Um die Lösung in einer anderen AWS-Region zu starten, verwenden Sie die Regionsauswahl in der Navigationsleiste der Konsole.

   Hinweis: Diese Lösung verwendet Amazon Kendra und Amazon Bedrock, die derzeit nicht in allen AWS-Regionen verfügbar sind. Wenn Sie diese Funktionen verwenden, müssen Sie diese Lösung in einer AWS-Region starten, in der diese Services verfügbar sind. Die aktuelle Verfügbarkeit nach Regionen finden Sie in der Liste der regionalen AWS-Dienste.

3. Vergewissern Sie sich auf der Seite Stack erstellen *, dass sich die richtige Vorlagen-URL im Textfeld *Amazon S3-URL * befindet, und wählen Sie *Weiter aus.

4. Weisen Sie auf der Seite *Stack-Details angeben *Ihrem Lösungs-Stack einen Namen zu. Informationen zu Einschränkungen bei der Benennung von Zeichen finden Sie unter IAM- und STS-Grenzwerte im AWS Identity and Access Management-Benutzerhandbuch.
5. Überprüfen Sie unter Parameter die Parameter für diese Lösungsvorlage und ändern Sie sie nach Bedarf. Diese Lösung verwendet die folgenden Standardwerte.

UseCaseUUID	*<_Requires input_>*	36 Zeichen lang UUIDv4 , um diesen bereitgestellten Anwendungsfall innerhalb einer Anwendung zu identifiz ieren.
UseCaseConfigRecordKey	*<_Requires input_>*	Schlüssel, der dem Datensatz entspricht, der Konfigura tionen enthält, die der Chat-Anbieter Lambda zur Laufzeit benötigt. Der Datensatz in der Tabelle muss ein Schlüssel attribut haben, das diesem Wert entspricht, und ein Konfigurationsattribut, das die gewünschte Konfigura tion enthält. Dieser Datensatz wird von der Bereitste llungsplattform aufgefüll t, falls er verwendet wird. Für eigenständige Bereitste llungen dieses Anwendung sfalls ist ein manuell erstellte r Eintrag in der in definierten Tabelle UseCaseConfigTable Nameerforderlich.
UseCaseConfigTableName	*<_Requires input_>*	Der Stack liest die Konfigura tion aus der Tabelle mit

		diesem Namen als Schlüssel UseCaseConfigRecordKey

ExistingRestApiId	(Optionale Eingabe)	Bestehende API-Gateway-REST-API-ID, die verwendet werden soll. Falls nicht angegeben, wird eine neue API-Gateway-REST-API erstellt. Wird normalerweise bei der Bereitstellung über das Deployment-Dashboard bereitgestellt. Hinweis: Die Verwendung von APIs Existing kann dazu beitragen, die Duplizierung von Ressourcen zu reduziere n und die Verwaltung zu vereinfachen, APIs wenn Sie mehrere eigenständige Anwendungsfälle bereitste llen müssen. Bei der Bereitste llung vorhandener Daten APIs für einen eigenstän digen Anwendungsfall sind Sie dafür verantwortlich, sicherzustellen, dass die API mit den erforderlichen Routen und den erwartete n Modellen konfiguriert ist. Eine erforderliche vorkonfig urierte /details-Route (ruft Anwendungsfalldetails während des Chats ab) und optional eine /feedback -Route (falls FeedbackE nabledso eingestellt, dass sie die Erfassung von Feedback für LLM-Chat-

		Antworten ermöglicht) Yes müssen konfiguriert werden. Zusätzlich ExistingCognitoUserPoolIdund ExistingCognitoGroupPolicyTableNamemuss ebenfalls ExistingApiRootResourceIdangegeben werden.
ExistingApiRootResourceId	(Optionale Eingabe)	Bestehende API-Gateway-REST-API-Root-Ressourcen-ID, die verwendet werden soll. Die REST-API-Root-Ressourcen-ID kann von der AWS-Konsole abgerufen werden, indem Sie die Root-Ressource (/) im Abschnitt „Ressourcen“ der API auswählen. Die Ressource n-ID wird dann im Bereich mit den Ressourcendetails angezeigt. Sie können alternativ einen API-Aufruf zur Beschreibung Ihrer REST-API ausführen, um die Root-Ressourcen-ID zu ermitteln.
FeedbackEnabled	No	Wenn diese Option auf Nein gesetzt ist, hat der bereitgestellte Anwendungsfallstapel keinen Zugriff auf die Feedback-Funktion.

ExistingModelInfoTableName	(Optionale Eingabe)	DynamoDB-Tabellenname für die Tabelle, die Modellinformationen und Standardwerte enthält. Wird von der Bereitstellungsplattform verwendet. Wenn nicht angegeben, wird eine neue Tabelle erstellt, die die Standardwerte des Modells enthält.
DefaultUserEmail	`placeholder@example.com`	E-Mail-Adresse des Standardbenutzers für diesen Anwendungsfall. Ein Amazon Cognito Cognito-Benutzer für diese E-Mail wird erstellt, um auf den Anwendungsfall zuzugreifen. Wenn nicht angegeben, werden die Cognito-Gruppe und der Cognito-Benutzer nicht erstellt. Sie können die Gruppe auch verwenden`placeholder@example.com`, um die Gruppe zu erstellen, aber nicht den Benutzer. Informationen zur Einrichtung Ihres Benutzerpools finden Sie unter Manuelle Konfiguration des Benutzerpools.

ExistingCognitoUserPoolId	(Optionale Eingabe)	UserPoolIdeines vorhanden en Amazon Cognito Cognito-Benutzerpools, mit dem dieser Anwendungsfall authentifiziert wird. Wird normalerweise bei der Bereitstellung über das Deployment-Dashboard bereitgestellt, kann aber weggelassen werden, wenn dieser Anwendungsfall-Sta ck eigenständig bereitgestellt wird.
CognitoDomainPrefix	(Optionale Eingabe)	Geben Sie einen Wert ein, wenn Sie eine Domäne für den Cognito User Pool Client bereitstellen möchten. Wenn Sie keinen Wert angeben, generiert die Bereitstellung einen Wert.
ExistingCognitoUserPoolClie nt	(Optionale Eingabe)	Stellen Sie einen Benutzerp ool-Client (App Client) bereit, um einen vorhandenen zu verwenden. Wenn Sie keinen Benutzerpool-Client bereitste llen, wird ein neuer erstellt. Dieser Parameter kann nur angegeben werden, wenn eine vorhandene Benutzerp ool-ID angegeben wird.

ExistingCognitoGroupPolicyTableName	(Optionale Eingabe)	Name der DynamoDB-Tabelle, die Benutzergruppenrichtlinien enthält. Dies wird vom benutzerdefinierten Autorisierer für die API des Anwendungsfalls verwendet. In der Regel können Sie bei der Bereitstellung über die Bereitstellungsplattform eine Eingabe vornehmen, bei der eigenständigen Bereitstellung dieses Anwendungsfall-Stacks kann diese jedoch weggelassen werden.
RAGEnabled	`true`	Wenn auf true gesetzt, verwendet der bereitgestellte Anwendungsfallstapel den bereitgestellten Amazon Kendra Kendra-Index, der für die Bereitstellung von RAG-Funktionen erstellt wurde. Wenn auf gesetzt`false`, interagiert der Benutzer direkt mit dem LLM.
KnowledgeBaseType	Bedrock	Wissensdatenbanktyp, der für RAG verwendet werden soll. Nur gesetzt, wenn RAGEnabled es ist`true`. Kann Bedrock oder Kendra sein. Hinweis: Nur relevant, wenn es wahr RAGEnabledist.

ExistingKendraIndexId	(Optionale Eingabe)	Index-ID eines vorhandenen Kendra-Indexes, der für den Anwendungsfall verwendet werden soll. Wenn keiner angegeben ist und Kendra KnowledgeBaseTypeist, wird ein neuer Index für Sie erstellt. Hinweis: Nur relevant, wenn RAGEnabledist `true` und Knowledge BaseTypeist`Kendra`.
NewKendraIndexName	(Optionale Eingabe)	Name für den neuen Kendra-Index, der für diesen Anwendungsfall erstellt werden soll. Gilt nur, ExistingKendraIndexIdwenn nicht angegeben. Hinweis: Nur relevant, wenn RAGEnabledes wahr ist und Kendra Knowledge BaseTypeist.

NewKendraQueryCapacityUnits	0	Zusätzliche Abfragekapazitätseinheiten für den neuen Amazon Kendra Kendra-Index, die für diesen Anwendungsfall erstellt werden sollen. Gilt nur, ExistingKendraIndexIdwenn nicht angegeben, siehe CapacityUnitsConfiguration. Hinweis: Nur relevant, wenn RAGEnabledes ist `true` und Knowledge BaseTypeist`Kendra`.
NewKendraStorageCapacityUnits	`0`	Zusätzliche Speicherkapazitätseinheiten für den neuen Amazon Kendra Kendra-Index sollen für diesen Anwendungsfall erstellt werden. Gilt nur, wenn ExistingKendraIndexIdes nicht mitgeliefert wird, siehe CapacityUnitsConfiguration. Hinweis: Nur relevant, wenn RAGEnabledes ist `true` und Knowledge BaseTypeist`Kendra`.

NewKendraIndexEdition	(Optionale Eingabe)	Die Edition von Amazon Kendra, die für den neuen Amazon Kendra Kendra-Index verwendet werden soll, der für diesen Anwendungsfall erstellt werden soll. Gilt nur, wenn ExistingKendraIndexIdnicht im Lieferumfang enthalten, siehe Amazon Kendra Editions. Hinweis: Nur relevant, wenn RAGEnabledes ist `true` und Knowledge BaseTypeistKendra.
BedrockKnowledgeBaseId	(Optionale Eingabe)	ID der Bedrock-Wissensdatenbank, die in einem RAG-Anwendungsfall verwendet werden soll. Kann nicht angegeben werden, wenn ExistingKendraIndexIdoder angegeben NewKendra IndexNamewerden. Hinweis: Nur relevant, wenn RAGEnabledist `true` und Knowledge BaseTypeistBedrock.
VpcEnabled	No	Sollen die Stack-Ressourcen innerhalb einer VPC bereitgestellt werden.

CreateNewVpc	No	Wählen Sie ausYes, ob die Lösung eine neue VPC für Sie erstellen und für diesen Anwendungsfall verwenden soll. Hinweis: Nur relevant, wenn ja VpcEnabled. Yes
IPAMPoolId	(Optionale Eingabe)	Wenn Sie den CIDR-Bereich mithilfe von Amazon VPC IP Address Manager zuweisen möchten, geben Sie die zu verwendende IPAM-Pool-ID an. Hinweis: Nur relevant, wenn VpcEnabledist und istYes. CreateNewVpcNo
ExistingVpcId	(Optionale Eingabe)	VPC-ID einer vorhandenen VPC, die für den Anwendung sfall verwendet werden soll. Hinweis: Nur relevant, wenn VpcEnabledist Yes und CreateNewVpcist. No

ExistingPrivateSubnetIds	(Optionale Eingabe)	Durch Kommas getrennte Liste der Subnetze IDs vorhandener privater Subnetze, die für die Bereitstellung der Lambda-Funktion verwendet werden sollen. Hinweis: Nur relevant, wenn VpcEnabledist und ist. `Yes` CreateNewVpcNo
ExistingSecurityGroupIds	(Optionale Eingabe)	Durch Kommas getrennte Liste von Sicherheitsgruppen der vorhandenen VPC, die für die Konfiguration von Lambda-Funktionen verwendet werden sollen. Hinweis: Nur relevant, wenn VpcEnabledist und ist`Yes`. CreateNewVpcNo
VpcAzs	(Optionale Eingabe)	Durch Kommas getrennte Liste, AZs in welcher Subnetze von erstellt wurden VPCs Hinweis: Nur relevant, wenn VpcEnabledist `Yes` und CreateNewVpcist. `No`

UseInferenceProfile	No	Wenn das konfigurierte Modell Bedrock ist, können Sie angeben, ob Sie Bedrock Inference Profile verwenden . Dadurch wird sichergestellt, dass die erforderlichen IAM-Richtlinien während der Stack-Bereitstellung konfiguriert werden. Weitere Informationen finden Sie in der folgenden Datei - region-inference.html https://docs.aws.amazon.com/bedrock/ latest/userguide/cross
Stellen Sie die Benutzeroberfläche bereit	Ja	Wählen Sie die Option zur Bereitstellung der Frontend-Benutzeroberfläche für diese Bereitstellung aus. Wenn Sie Nein auswählen, wird nur die Infrastruktur für das APIs Hosten der Authentifizierung und die APIs Backend-Verarbeitung erstellt.

6. Wählen Sie Weiter aus.
7. Wählen Sie auf der Seite Configure stack options (Stack-Optionen konfigurieren) Next (Weiter) aus.
8. Überprüfen und bestätigen Sie die Einstellungen auf der Seite Review. Markieren Sie das Kästchen, um zu bestätigen, dass die Vorlage AWS Identity and Access Management (IAM) - Ressourcen erstellt.
9. Wählen Sie Stack erstellen aus, um den Stack bereitzustellen.

Sie können den Status des Stacks in der CloudFormation AWS-Konsole in der Spalte Status anzeigen. Sie sollten in etwa 10 bis 30 Minuten den Status CREATE_COMPLETE erhalten.

Bereitstellung eines eigenständigen Bedrock Agent-Anwendungsfalls

Folgen Sie den step-by-step Anweisungen in diesem Abschnitt, um die Lösung zu konfigurieren und in Ihrem Konto bereitzustellen.

Zeit bis zur Bereitstellung: Ungefähr 10-30 Minuten

1. Melden Sie sich bei der AWS-Managementkonsole an und klicken Sie auf die Schaltfläche, um die CloudFront Vorlage zu starten.

BedrockAgent.vorlage	

2. Die Vorlage wird standardmäßig in der Region USA Ost (Nord-Virginia) gestartet. Um die Lösung in einer anderen AWS-Region zu starten, verwenden Sie die Regionsauswahl in der Navigationsleiste der Konsole.

> **Note**
>
> Diese Lösung verwendet Amazon Bedrock, das derzeit nicht in allen AWS-Regionen verfügbar ist. Wenn Sie diese Funktionen verwenden, müssen Sie diese Lösung in einer AWS-Region starten, in der diese Services verfügbar sind. Die aktuelle Verfügbarkeit nach Regionen finden Sie in der Liste der regionalen AWS-Dienste.

3. Vergewissern Sie sich auf der Seite Stack erstellen, dass sich die richtige Vorlagen-URL im Textfeld Amazon S3 S3-URL befindet, und wählen Sie Weiter.
4. Weisen Sie Ihrem Lösungsstapel auf der Seite „Stack-Details angeben“ einen Namen zu. Informationen zu Einschränkungen bei der Benennung von Zeichen finden Sie unter {https---docs-aws-amazon-com- https---docs-aws-amazon-com -IAM-Latest- UserGuide -reference-iam-limits-html} [IAM- und AWS STS STS-Kontingente] im AWS Identity and Access Management-Benutzerhandbuch.
5. Überprüfen Sie unter Parameter die Parameter für diese Lösungsvorlage und ändern Sie sie nach Bedarf. Diese Lösung verwendet die folgenden Standardwerte.

Parameter	Standardeintrag	Description
UseCaseUUID	*<_Requires input_>*	36 Zeichen lang UUIDv4 , um diesen bereitgestellten Anwendungsfall innerhalb einer Anwendung zu identifiz ieren.
UseCaseConfigRecordKey	*<Requires input>*	Schlüssel, der dem Datensatz entspricht, der Konfigura tionen enthält, die von der Lambda-Funktion des Chat-Anbieters zur Laufzeit benötigt werden. Der Datensatz in der Tabelle muss ein Schlüssel attribut haben, das diesem Wert entspricht, und ein Konfigurationsattribut, das die gewünschte Konfiguration enthält. Dieser Datensatz wird von der Bereitstellungsplattform aufgefüllt, falls sie verwendet wird. Für eigenständige Bereitstellungen dieses Anwendungsfalls ist ein manuell erstellter Eintrag in der in definierten Tabelle UseCaseConfigTable Nameerforderlich.

Parameter	Standardeintrag	Description
UseCaseConfigTableName	*<Requires input>*`	Der Stack liest die Anwendungsfallkonfiguration aus der hier bereitgestellten Tabelle und verwendet dabei den in UseCaseConfigRecordKeydefinierten Datensatzschlüssel.
DefaultUserEmail	placeholder@example.com	E-Mail des Standardbenutzers für diesen Anwendungsfall. Die Lösung erstellt einen Amazon Cognito Cognito-Benutzer für diese E-Mail, um auf den Anwendungsfall zuzugreifen.

Parameter	Standardeintrag	Description
ExistingRestApiId	(Optionale Eingabe)	Bestehende API-Gateway-REST-API-ID, die verwendet werden soll. Falls nicht angegeben, wird eine neue API-Gateway-REST-API erstellt. Wird normalerweise bei der Bereitstellung über das Deployment-Dashboard bereitgestellt. Hinweis: Die Verwendung von APIs Existing kann dazu beitragen, die Duplizierung von Ressourcen zu reduziere n und die Verwaltung zu vereinfachen, APIs wenn Sie mehrere eigenständige Anwendungsfälle bereitste llen müssen. Bei der Bereitste llung vorhandener Daten APIs für einen eigenstän digen Anwendungsfall sind Sie dafür verantwortlich, sicherzustellen, dass die API mit den erforderlichen Routen und den erwartete n Modellen konfiguriert ist. Eine erforderliche vorkonfig urierte /details-Route (ruft Anwendungsfalldetails während des Chats ab) und optional eine /feedback -Route (falls FeedbackE nabledso eingestellt, dass sie die Erfassung von

Parameter	Standardeintrag	Description
		Feedback für LLM-Chat-Antworten ermöglicht) Yes müssen konfiguriert werden. Zusätzlich ExistingCognitoUserPoolIdund ExistingCognitoGroupPolicyTableNamemuss ebenfalls ExistingApiRootResourceIdangegeben werden.
ExistingApiRootResourceId	(Optionale Eingabe)	Bestehende API-Gateway-REST-API-Root-Ressourcen-ID, die verwendet werden soll. Die REST-API-Root-Ressourcen-ID kann von der AWS-Konsole abgerufen werden, indem Sie die Root-Ressource (/) im Abschnitt „Ressourcen“ der API auswählen. Die Ressource n-ID wird dann im Bereich mit den Ressourcendetails angezeigt. Sie können alternativ einen API-Aufruf zur Beschreibung Ihrer REST-API ausführen, um die Root-Ressourcen-ID zu ermitteln.
FeedbackEnabled	No	Wenn diese Option auf Nein gesetzt ist, hat der bereitgestellte Anwendungsfallstapel keinen Zugriff auf die Feedback-Funktion.

Parameter	Standardeintrag	Description
CognitoDomainPrefix	(Optionale Eingabe)	Geben Sie einen Wert ein, wenn Sie eine Domain für den Amazon Cognito Cognito-Benutzerpool-Client bereitstellen möchten. Wenn Sie keinen Wert angeben, generiert die Lösung einen.
ExistingCognitoUserPoolId	(Optionale Eingabe)	UserPoolIdeines vorhanden en Amazon Cognito Cognito-Benutzerpools, mit dem Sie diesen Anwendungsfall authentifizieren möchten. HINWEIS: In der Regel geben Sie diese ID bei der Bereitstellung über das Bereitstellungs-Dashboard an. Sie können sie jedoch weglassen, wenn Sie diesen Anwendungsfall-Stack eigenständig bereitstellen.
ExistingCognitoUserPoolClie nt	(Optionale Eingabe)	Stellen Sie einen Benutzerp ool-Client (App-Client) bereit, um einen vorhandenen zu verwenden. Wenn Sie keinen Benutzerpool-Client bereitste llen, erstellt die Lösung einen. Sie können diesen Parameter nur angeben, wenn Sie einen angegeben haben ExistingC ognitoUserPoolId.

Parameter	Standardeintrag	Description
ExistingCognitoGroupPolicyTableName	(Optionale Eingabe)	Name der DynamoDB-Tabelle, die Benutzergruppenrichtlinien enthält. Dies wird vom benutzerdefinierten Autorisierer für die API des Anwendungsfalls verwendet. HINWEIS: Normalerweise geben Sie diesen Namen bei der Bereitstellung über das Deployment-Dashboard an. Sie können ihn jedoch weglassen, wenn Sie diesen Anwendungsfall-Stack eigenständig bereitstellen.
VpcEnabled	`No`	Ob die Stack-Ressourcen innerhalb einer VPC bereitgestellt werden.
CreateNewVpc	`No`	Wählen Sie aus, `Yes` ob die Lösung eine neue VPC für Sie erstellen und für diesen Anwendungsfall verwenden soll. HINWEIS: Dieser Parameter ist nur relevant, wenn er relevant VpcEnable distYes.

Parameter	Standardeintrag	Description
IPAMPoolId	(Optionale Eingabe)	Wenn Sie den CIDR-Bereich mithilfe von IPAM zuweisen möchten, geben Sie die zu verwendende IPAM-Pool -ID an. HINWEIS: Dieser Parameter ist nur relevant, wenn er und VpcEnable distYes. CreateNewVpcNo
ExistingVpcId	(Optionale Eingabe)	VPC-ID einer vorhandenen VPC, die für den Anwendung sfall verwendet werden soll. HINWEIS: Dieser Parameter ist nur relevant, wenn er Yes und VpcEnabledCreateNew Vpcist. No
ExistingPrivateSubnetIds	(Optionale Eingabe)	Durch Kommas getrennte Liste der Subnetze IDs vorhandener privater Subnetze, die für die Bereitstellung der Lambda-Funktion verwendet werden sollen. HINWEIS: Dieser Parameter ist nur relevant, wenn VpcEnableder ist und ist. Yes CreateNewVpcNo

Parameter	Standardeintrag	Description
ExistingSecurityGroupIds	(Optionale Eingabe)	Durch Kommas getrennte Liste der Sicherheitsgruppen der vorhandenen VPC, die für die Konfiguration von Lambda-Funktionen verwendet werden sollen. HINWEIS: Dieser Parameter ist nur relevant, wenn VpcEnableder und ist. `Yes` CreateNewVpcNo
VpcAzs	(Optionale Eingabe)	Durch Kommas getrennte Liste, AZs in welcher Subnetze von erstellt wurden VPCs Hinweis: Nur relevant, wenn VpcEnabledist `Yes` und CreateNewVpcist. `No`
BedrockAgentId	*`<Requires input>`*	Die ID des Amazon Bedrock Agents, der verwendet werden soll.
BedrockAgentAliasId	*`<Requires input>`*	Die Alias-ID des Amazon Bedrock Agents, der verwendet werden soll.

Parameter	Standardeintrag	Description
Stellen Sie die Benutzero berfläche bereit	`Yes`	Wählen Sie die Option zur Bereitstellung der Frontend-Chat-Benutzeroberfläche für diese Bereitstellung aus. Die Auswahl No führt zur Erstellung der Infrastruktur für das Hosten der APIs, der Authentifizierung für die APIs und der Backend-V erarbeitung ohne die Chat-Benutzeroberfläche.

6. Wählen Sie Weiter aus.
7. Wählen Sie auf der Seite Configure stack options (Stack-Optionen konfigurieren) Next (Weiter) aus.
8. Überprüfen und bestätigen Sie die Einstellungen auf der Seite Review. Aktivieren Sie das Kästchen zur Bestätigung, dass die Vorlage IAM-Ressourcen erstellt.
9. Wählen Sie Stack erstellen aus, um den Stack bereitzustellen.

Sie können den Status des Stacks in der CloudFormation AWS-Konsole in der Spalte Status anzeigen. Sie sollten in etwa 10 bis 30 Minuten den Status CREATE_COMPLETE erhalten.

Bereitstellung einer DynamoDB-Chat-Konfiguration

Bei der Bereitstellung eines Anwendungsfalls UseCaseConfigTableNamesind CloudFormation Parameter erforderlich, UseCaseConfigRecordKeydie normalerweise vom Deployment-Dashboard aufgefüllt werden. Der Stack der Bereitstellungs-Dashboards kümmert sich um die Erstellung und Konfiguration dieser Tabelle, während Aufrufe der Deployment-API das Auffüllen der Parameter auslösen.

Wenn Sie eine eigenständige Bereitstellung durchführen, müssen Sie wie folgt vorgehen:

1. Erstellen Sie eine DynamoDB-Tabelle mit einem Hash-Schlüssel oder Schlüssel.

2. Erstellen Sie einen Datensatz in der Tabelle, der die Konfiguration für den Anwendungsfall enthält, als Datensatz des folgenden Formats: `{key: some_use_case_key, config: {your_configuration}.`
3. Übergeben Sie bei der Bereitstellung die ausgewählten Parameter UseCaseConfigTableNameund UseCaseConfigRecordKey(`some_use_case_key`in diesem Beispiel) an den Anwendungsfallstapel.

Um eine geeignete Konfiguration für eine eigenständige Bereitstellung zu erstellen, können Sie im Deployment-Dashboard einen erforderlichen Anwendungsfall erstellen und den Datensatz aus der Konfigurationstabelle kopieren. Andernfalls können Sie anhand des folgenden Beispiels für eine Bedrock-Bereitstellung Ihre eigene Konfiguration erstellen:

```
{
 "UseCaseName": "SampleUseCase",
 "ConversationMemoryParams": {
 "ConversationMemoryType": "DynamoDB",
 "HumanPrefix": "H",
 "AiPrefix": "A",
 "ChatHistoryLength": 20
 },
 "KnowledgeBaseParams": {
 "KnowledgeBaseType": "Bedrock",
 "NumberOfDocs": 2,
 "ScoreThreshold": 0,
 "ReturnSourceDocs": false,
 "BedrockKnowledgeBaseParams": {
 "BedrockKnowledgeBaseId": "SOME_ID",
 "OverrideSearchType": null
 }
 },
 "LlmParams": {
 "ModelProvider": "Bedrock",
 "BedrockLlmParams": { "ModelId": "anthropic.claude-v2" },
 "PromptParams": {
 "PromptTemplate": "some prompt",
 "MaxPromptTemplateLength": 187500,
 "MaxInputTextLength": 187500,
 "UserPromptEditingEnabled": true,
 "DisambiguationEnabled": true,
 "DisambiguationPromptTemplate": "some prompt"
 },
```

```
  "ModelParams": {},
  "Temperature": 1,
  "RAGEnabled": true,
  "Streaming": true,
  "Verbose": false
  }
}
```

Überwachen Sie die Lösung mit Service Catalog AppRegistry

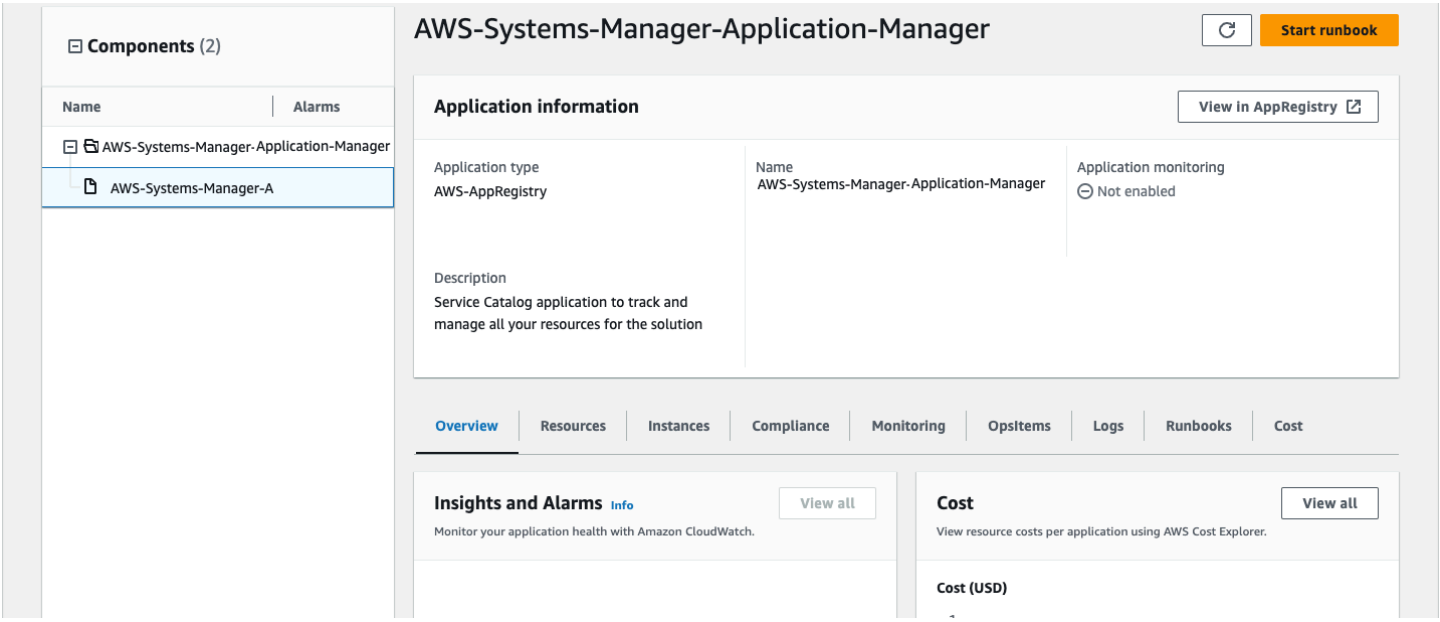
Die Lösung umfasst eine Service AppRegistry Catalog-Ressource, mit der die CloudFormation Vorlage und die zugrunde liegenden Ressourcen als Anwendung sowohl in Service Catalog AppRegistry als auch im Systems Manager Application Manager registriert werden können.

Systems Manager Application Manager bietet Ihnen einen Überblick über diese Lösung und ihre Ressourcen auf Anwendungsebene, sodass Sie:

- Überwachen Sie die Ressourcen, die Kosten für die bereitgestellten Ressourcen in allen Stacks und AWS-Konten sowie die mit dieser Lösung verknüpften Protokolle von einem zentralen Standort aus.
- Zeigen Sie Betriebsdaten für die Ressourcen dieser Lösung im Kontext einer Anwendung an. Zum Beispiel Bereitstellungsstatus, CloudWatch Alarme, Ressourcenkonfigurationen und Betriebsprobleme.

Die folgende Abbildung zeigt ein Beispiel für die Anwendungsansicht für den Lösungsstapel in Application Manager.

Stellt den Lösungsstapel in Application Manager dar

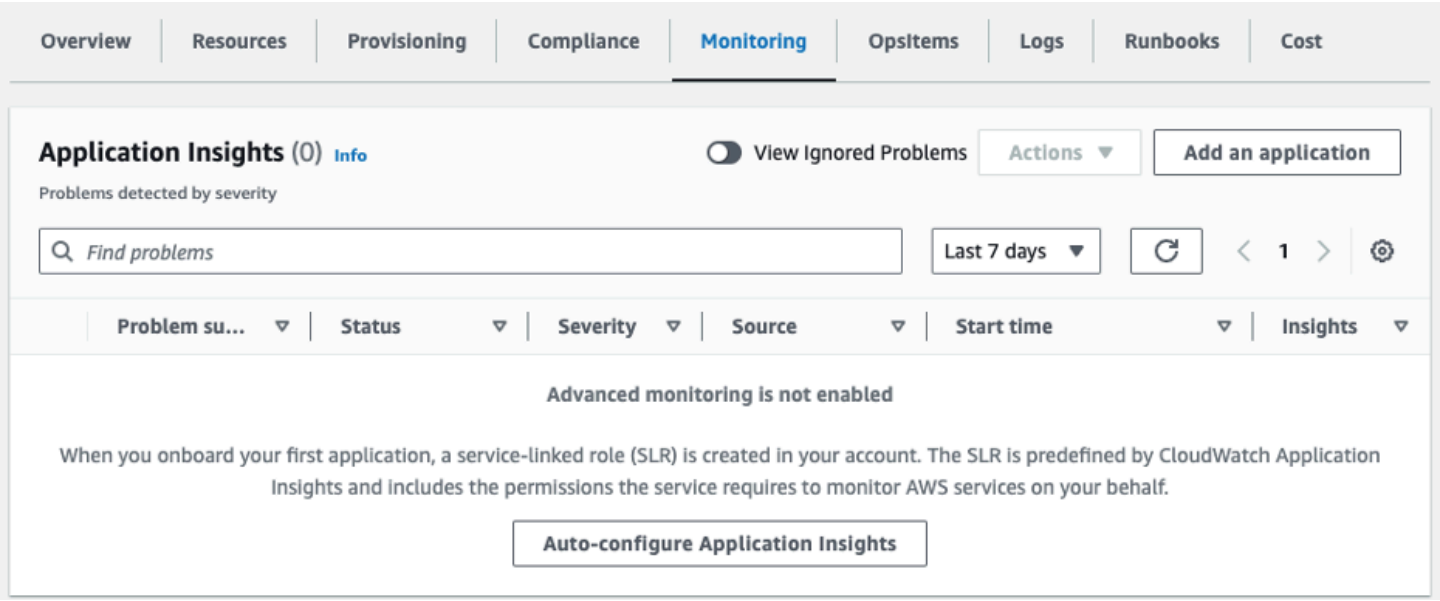


Aktivieren Sie Application Insights CloudWatch

1. Melden Sie sich bei der Systems Manager Manager-Konsole an.
2. Wählen Sie im Navigationsbereich Application Manager aus.
3. Suchen Sie unter Anwendungen nach dem Anwendungsnamen für diese Lösung und wählen Sie ihn aus.

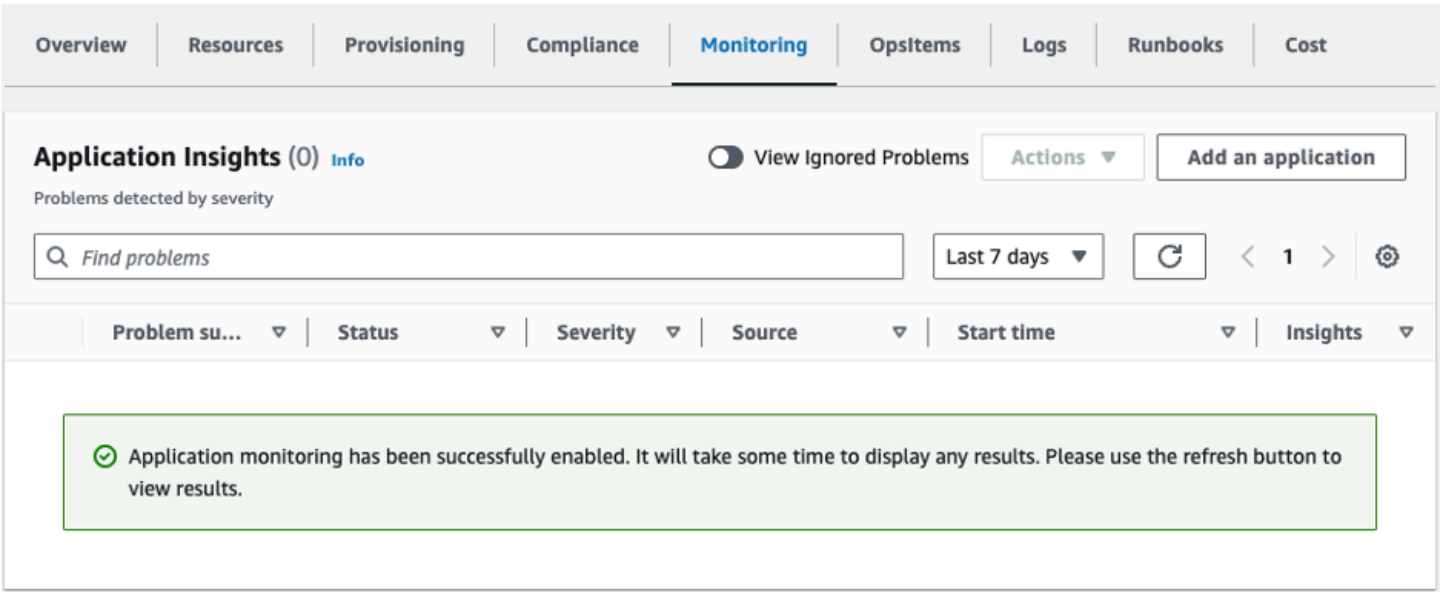
   Der Anwendungsname wird in der Spalte Anwendungsquelle den Eintrag App Registry haben und eine Kombination aus Lösungsname, Region, Konto-ID oder Stackname enthalten.
4. Wählen Sie in der Komponentenstruktur den Anwendungsstapel aus, den Sie aktivieren möchten.
5. Wählen Sie auf der Registerkarte Überwachung unter Application Insights die Option Application Insights automatisch konfigurieren aus.

Das Application Insights-Dashboard zeigt keine erkannten Probleme und die Option zur automatischen Konfiguration an.



Die Überwachung Ihrer Anwendungen ist jetzt aktiviert und das folgende Statusfeld wird angezeigt:

Das Application Insights-Dashboard zeigt die Meldung zur erfolgreichen Aktivierung der Überwachung an.



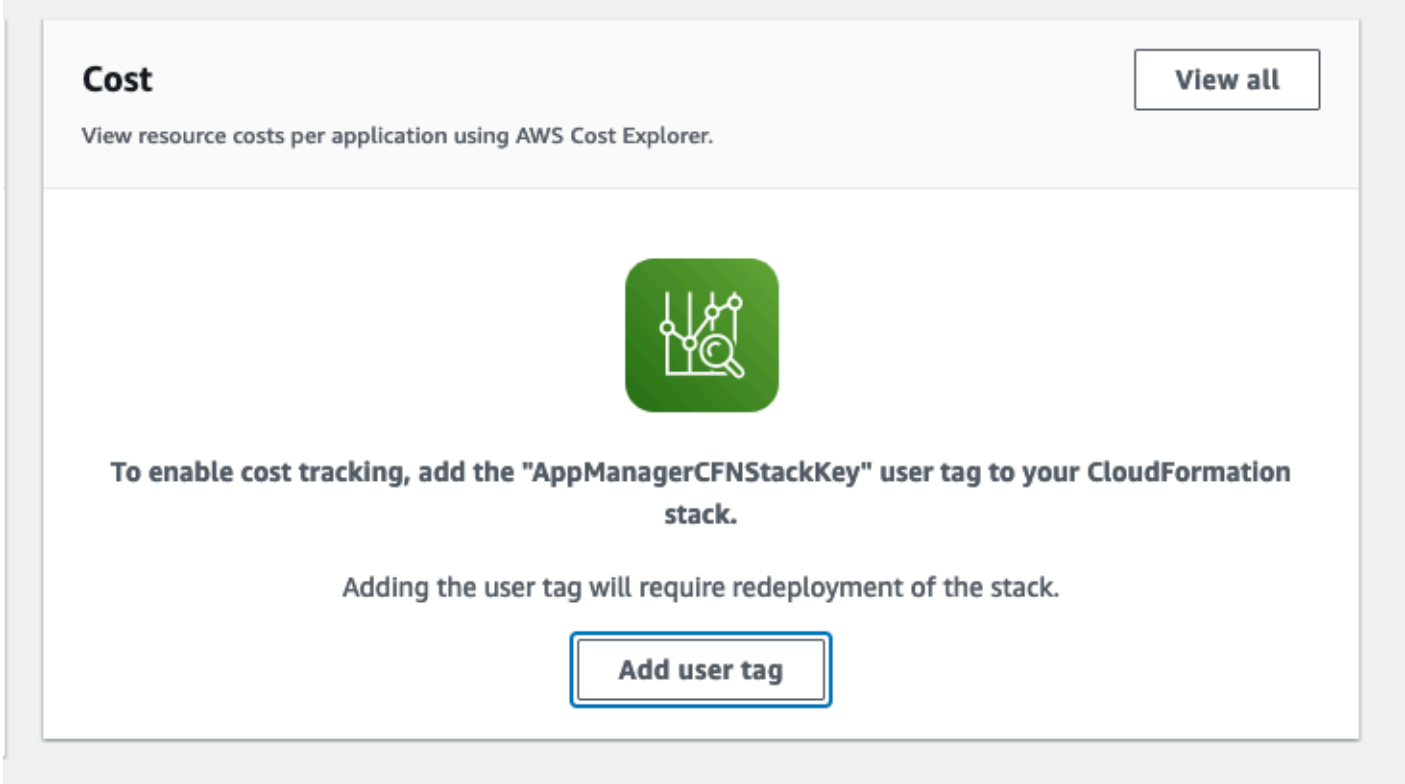
Bestätigen Sie die mit der Lösung verknüpften Kostenangaben

Nachdem Sie die mit der Lösung verknüpften Kostenzuordnungs-Tags aktiviert haben, müssen Sie die Kostenzuordnungs-Tags bestätigen, um die Kosten für diese Lösung zu sehen. So bestätigen Sie die Tags für die Kostenzuweisung:

1. Melden Sie sich bei der Systems Manager Manager-Konsole an.
2. Wählen Sie im Navigationsbereich Application Manager aus.
3. Wählen Sie unter Anwendungen den Anwendungsnamen für diese Lösung und wählen Sie ihn aus.

   Der Anwendungsname wird in der Spalte Anwendungsquelle den Eintrag App Registry haben und eine Kombination aus Lösungsname, Region, Konto-ID oder Stackname enthalten.
4. Wählen Sie auf der Registerkarte Übersicht unter Kosten die Option Benutzertag hinzufügen aus.

   Screenshot, der den Bildschirm „Anwendungskosten — Benutzertag hinzufügen“ zeigt



5. Geben Sie auf der Seite „Benutzertag hinzufügen“ den Text ein `confirm` und wählen Sie dann Benutzertag hinzufügen aus.

Es kann bis zu 24 Stunden dauern, bis der Aktivierungsvorgang abgeschlossen ist und die Tag-Daten angezeigt werden.

Aktivieren Sie die mit der Lösung verknüpften Kostenzuweisungs-Tags

Nachdem Sie den Cost Explorer aktiviert haben, müssen Sie die mit dieser Lösung verknüpften Kostenzuordnungs-Tags aktivieren, um die Kosten für diese Lösung zu sehen. Die Kostenzuweisungs-Tags können nur über das Verwaltungskonto der Organisation aktiviert werden. So aktivieren Sie Tags für die Kostenzuweisung:

1. Melden Sie sich bei der AWS Billing and Cost Management and Cost Management-Konsole an.
2. Wählen Sie im Navigationsbereich die Option Cost Allocation Tags aus.
3. Filtern Sie auf der Seite mit den Tags für die Kostenzuweisung nach dem AppManager CFNStack Schlüssel-Tag und wählen Sie dann das Tag aus den angezeigten Ergebnissen aus.

4. Wählen Sie Aktivieren.

AWS Cost Explorer

Durch die Integration mit dem AWS Cost Explorer, der zuerst aktiviert werden muss, können Sie sich in der Application Manager-Konsole einen Überblick über die mit der Anwendung und den Anwendungskomponenten verbundenen Kosten anzeigen lassen. Der Cost Explorer hilft Ihnen bei der Kostenverwaltung, indem er Ihnen einen Überblick über Ihre AWS-Ressourcenkosten und die Nutzung im Laufe der Zeit bietet. So aktivieren Sie den Cost Explorer für die Lösung:

1. Melden Sie sich bei der AWS Cost Management-Konsole an.
2. Wählen Sie im Navigationsbereich Cost Explorer aus, um die Kosten und die Nutzung der Lösung im Zeitverlauf anzuzeigen.

Aktualisieren Sie die Lösung

Wenn Sie die Lösung bereits bereitgestellt haben, gehen Sie wie folgt vor, um den CloudFormation Lösungsstapel zu aktualisieren und die neuesten Funktionen und Verbesserungen zu erhalten. Der Upgrade-Prozess besteht aus drei Teilen:

- Schritt 1: Bereitstellungs-Dashboard aktualisieren
- Schritt 2: Migrieren Sie Anwendungsfallkonfigurationen
- Schritt 3: Anwendungsfälle aktualisieren

> **Note**
>
> 1. In Version 2.0.0 wurde die Integration mit Anthropic und Hugging Face zugunsten von Amazon Bedrock und Amazon AI eingestellt. SageMaker Sie können Modelle, die über Hugging Face verfügbar sind, über bereitstellen. SageMaker JumpStart Weitere Informationen finden Sie unter Use Hugging Face with Amazon SageMaker AI.
> 2. Stellen Sie sicher, dass Sie den Aktualisierungsprozess in einer Umgebung außerhalb der Produktionsumgebung testen, bevor Sie diese Schritte ausführen.

Schritt 1: Bereitstellungs-Dashboard aktualisieren

1. Melden Sie sich bei der CloudFormation Konsole an, wählen Sie Ihren vorhandenen CloudFormation Stack aus und wählen Sie Aktualisieren aus.
2. Wählen Sie Aktuelle Vorlage ersetzen aus.
3. Gehen Sie unter Vorlage angeben wie folgt vor:
   a. Wählen Sie Amazon S3 S3-URL aus.
   b. Kopieren Sie den neuesten CloudFormation Vorlagenlink.
   c. Fügen Sie den Link in das Amazon S3 S3-URL-Feld ein.
   d. Vergewissern Sie sich, dass die richtige Vorlagen-URL im Textfeld Amazon S3 S3-URL angezeigt wird, und wählen Sie Weiter. Wählen Sie erneut Next (Weiter).

4. Überprüfen Sie unter Parameter die Parameter für die Vorlage und ändern Sie sie nach Bedarf. Einzelheiten zu den Parametern finden Sie unter Schritt 1: Starten des Deployment-Dashboard-Stacks.
5. Wählen Sie Weiter aus.
6. Wählen Sie auf der Seite Configure stack options (Stack-Optionen konfigurieren) Next (Weiter) aus.
7. Überprüfen und bestätigen Sie die Einstellungen auf der Seite Review. Markieren Sie das Kästchen, um zu bestätigen, dass die Vorlage IAM-Ressourcen erstellt.
8. Wählen Sie „Änderungssatz anzeigen“ und überprüfen Sie die Änderungen.
9. Wählen Sie Stack aktualisieren, um den Stack bereitzustellen.

Sie können den Status des Stacks in der CloudFormation AWS-Konsole in der Spalte Status anzeigen. Sie sollten in etwa 10 Minuten den Status UPDATE_COMPLETE erhalten.

Wenn die bestehende Lösungsversion älter als v2.0.0 war, erstellt das Update einen Web-UI-Stack (der die `amplify-ui` Implementierung des Anmeldebildschirms durch eine von Cognito gehostete Benutzeroberfläche ersetzt) und eine neue CloudFront URL, die im Output-Bereich der CloudFormation Konsole abgerufen werden kann, sobald der Stack-Status UPDATE_COMPLETE lautet.

> **Note**
>
> Bestehende Anwendungsfälle, die mit Versionen vor v2.0.0 erstellt wurden, werden ERST angezeigt, wenn Sie die unten beschriebenen Schritte ausgeführt haben.

Schritt 2: Migrieren Sie Anwendungsfallkonfigurationen (nur Updates von Versionen unter 2.0.0)

Das Schema für die Speicherung und die Konfiguration des AWS-Service zum Speichern von Anwendungsfällen wurden in Version 2.0.0 geändert. Folgen Sie den im GAAB v2-Migrationsbenutzerhandbuch beschriebenen Schritten mithilfe des Skripts gaab_v2_migration.py. Nachdem Sie das Skript ausgeführt haben, können Sie auf das Deployment-Dashboard zugreifen, um die bereitgestellten Anwendungsfälle anzuzeigen.

> **Note**
>
> Sie müssen die folgenden Schritte ausführen, um die Migration der Anwendungsfälle abzuschließen.

Schritt 3: Anwendungsfälle aktualisieren

Sie können die bereitgestellten Anwendungsfälle mit neuen Funktionen bearbeiten, die in den neuesten Versionen von GAAB verfügbar sind. Informationen zur Verwendung der Funktionen in dieser Lösung finden Sie unter Verwenden der Lösung.

Um Anwendungsfälle auf die neueste Version zu aktualisieren, müssen Sie die Schritte „Anwendungsfall bearbeiten“ im Bereitstellungs-Dashboard ausführen (obwohl Sie möglicherweise keine Änderungen vornehmen). Diese Aktion löst ein CloudFormation Stack-Update mit der neuesten Vorlagenversion aus.

> **Note**
>
> Anwendungsfälle, die mit 1.x- oder 2.x-Versionen der Lösung erstellt wurden, funktionieren möglicherweise nicht mit späteren Versionen. Daher empfehlen wir, bestehende Anwendungsfälle, die mit Versionen vor Version 3.0.0 erstellt wurden, über das Deployment-Dashboard zu klonen. Migrieren Sie dann schrittweise und ersetzen Sie sie durch neue Anwendungsfälle, die mit Version 3.0.0 oder höher erstellt wurden.

Fehlerbehebung

Dieser Abschnitt enthält Anweisungen zur Fehlerbehebung für die Bereitstellung und Verwendung der Lösung.

Wenn diese Anweisungen Ihr Problem nicht lösen, erhalten Sie von Contact Support Anweisungen zum Öffnen einer Support-Anfrage für diese Lösung.

Problem: Die Bereitstellung einer VPC-fähigen Konfiguration mit Create a VPC for me schlägt fehl

Der Deployment-Dashboard-Stack oder der Use Case-Stack schlägt bei der Bereitstellung fehl, weil der CloudFormation keine VPC-Netzwerkressourcen bereitstellen konnte.

Auflösung

Überprüfen Sie die Kontingentgrenzen für VPCs und Elastic IPs in Ihrem Konto. Die Standardlimits sind jeweils 5 für Elastic IPs und VPCs pro AWS-Konto pro AWS-Region.

> **Note**
>
> Wenn die Lösung eine VPC erstellt, ist eine einzelne VPC-fähige Bereitstellung (Deployment Dashboard oder Use Case) eine 2-AZ-Bereitstellung mit einem öffentlichen und einem privaten Subnetz in jeder AZ. Jedes öffentliche Subnetz stellt 1 NAT-Gateway bereit.
> Bei 2 NAT-Gateways verbraucht die Bereitstellung 2 öffentliche IP-Adressen aus dem Kontingentlimit.

Einige Einschränkungen, die Sie beachten sollten (pro Konto, pro Region):

- Anzahl von VPCs — 5
- Anzahl der öffentlichen IP-Adressen: 5
- Anzahl der Gateway-VPC-Endpunkte: 20
- Anzahl der VPC-Endpunkte mit Schnittstelle: 20

Problem: Der Anwendungsfall-Stack kann nicht gelöscht werden, CloudFormation nachdem der Deployment-Dashboard-Stack gelöscht wurde

Wenn der Deployment-Dashboard-Stapel gelöscht wird, CloudFormation bevor alle Anwendungsfall-Stapel gelöscht wurden, können die Anwendungsfälle in einen gesperrten (unbrauchbaren) Zustand geraten. Dies liegt daran, dass eine vom Deployment-Dashboard-Stack erstellte IAM-Rolle nicht mehr existiert, sodass Änderungen am Anwendungsfallstapel verhindert werden.

Auflösung

> **Warning**
>
> Stellen Sie sicher, dass Sie alle manuell erstellten Rollen sofort nach der Verwendung bereinigen. Dabei handelt es sich um erweiterte Berechtigungen, die Benutzer für die Erhöhung ihrer Rollen nutzen können.

Erstellen Sie die gelöschte IAM-Rolle neu, um das Löschen der CloudFormation Stacks zu ermöglichen:

1. Öffnen Sie die CloudFormation Konsole und ermitteln Sie die Rolle, die Ihrem gesperrten Stack zugeordnet ist.
   a. Den Rollen-ARN finden Sie im Abschnitt mit den Stack-Informationen mit der Bezeichnung IAM-Rolle.
   b. Der Rollenname folgt nach:role/ im ARN der IAM-Rolle (z. B. arn:aws:iam: :role/) <account-id><role-name>
2. Erstellen Sie in IAM eine neue Rolle mit demselben Namen wie die gelöschte Rolle.
   a. Wählen Sie den AWS-Service als vertrauenswürdige Entität aus und wählen Sie ihn CloudFormationaus der Drop-down-Liste aus.
   b. Fügen Sie die erforderlichen Berechtigungen hinzu. Wenn Sie sich nicht sicher sind, welche Berechtigungen erforderlich sind, können Sie die von AWS verwaltete AdministratorAccessRichtlinie verwenden.
   c. Geben Sie den Rollennamen genau so ein, wie Sie ihn in Schritt 1 erhalten haben.
3. Kehren Sie zur CloudFormation Konsole zurück und löschen Sie die gesperrten Stacks.

4. Sobald alle gesperrten Stacks erfolgreich gelöscht wurden, kehren Sie zu IAM zurück und löschen Sie alle in Schritt 2 erstellten Rollen.

Problem: Die Benutzeroberfläche für Anwendungsfälle spiegelt keine Änderungen an den Einstellungen wider

Wenn Anwendungsfälle aktualisiert werden, wird die Benutzeroberfläche für bereitgestellt CloudFront. Da jedoch sowohl Bereitstellungen als auch die Konfigurationsdatei CloudFront zwischengespeichert werden, die vorgibt, wie einige Einstellungen dem Benutzer angezeigt werden, werden diese Änderungen möglicherweise nicht sofort übernommen.

Auflösung

Die CloudFront Verteilung kann ungültig gemacht werden, um zu erzwingen, dass die neue Konfiguration an Frontend-Benutzer weitergegeben wird.

1. Öffnen Sie die CloudFormation Konsole und ermitteln Sie die CloudFront Distribution, die Ihrem Anwendungsfall-Stack zugeordnet ist.
   a. Der Anwendungsfallstapel sollte mit demselben Namen beginnen, den Sie bei der Bereitstellung des Anwendungsfalls verwendet haben.
   b. Suchen Sie den verschachtelten Stack, der der Benutzeroberfläche entspricht. Der Name des verschachtelten Stacks sollte mit WebAppS3 StackS3 UINested beginnen. UINested StackResource
   c. Suchen Sie auf der Registerkarte Ressourcen nach der Ressource des Typs AWS::CloudFront::Distributionund wählen Sie dann die physische ID aus. Dadurch wird die Distribution in der CloudFront Konsole geöffnet.
2. Navigieren Sie zur Registerkarte Invalidierungen, wählen Sie dann Create Invalidation aus und geben Sie den Pfad /* ein. Dadurch werden alle Pfade ungültig.
3. Löschen Sie in Ihrem eigenen Browser alle Cookies und zwischengespeicherten Dateien, die sich auf den Anwendungsfall beziehen.

Kontaktieren Sie AWS Support.

Wenn Sie über AWS Business Support+, AWS Enterprise Support oder Unified Operations verfügen, können Sie das AWS Support Center nutzen, um fachkundige Support zu dieser Lösung zu erhalten. In den folgenden Abschnitten finden Sie entsprechende Anweisungen.

Fall erstellen

1. Melden Sie sich im Support Center an.
2. Wählen Sie Create case (Fall erstellen) aus.

Wie können wir helfen?

1. Wählen Sie Technisch.
2. Wählen Sie für Service die Option Lösungen aus.
3. Wählen Sie als Kategorie die Option Andere Lösungen aus.
4. Wählen Sie unter Schweregrad die Option aus, die Ihrem Anwendungsfall am besten entspricht.
5. Wenn Sie den Service, die Kategorie und den Schweregrad eingeben, werden in der Benutzeroberfläche Links zu häufig gestellten Fragen zur Fehlerbehebung angezeigt. Wenn Sie Ihre Frage mit diesen Links nicht lösen können, wählen Sie Nächster Schritt: Zusätzliche Informationen.

Zusätzliche Informationen

1. Geben Sie als Betreff einen Text ein, der Ihre Frage oder Ihr Problem zusammenfasst.
2. Beschreiben Sie als Beschreibung das Problem detailliert, einschließlich des Namens dieser Lösung: Generative AI Application Builder on AWS.
3. Wählen Sie Dateien anhängen.
4. Fügen Sie die Informationen bei, die der AWS-Support zur Bearbeitung der Anfrage benötigt.

Helfen Sie uns, Ihren Fall schneller zu lösen

1. Geben Sie die angeforderten Informationen ein.

2. Klicken Sie auf Next step: Solve now or contact us (()Nächster Schritt): Jetzt lösen oder Support kontaktieren).

Löse es jetzt oder kontaktiere uns

1. Sehen Sie sich die Solve Now-Lösungen an.
2. Wenn Sie Ihr Problem mit diesen Lösungen nicht lösen können, wählen Sie Kontaktieren Sie uns, geben Sie die angeforderten Informationen ein und klicken Sie auf Absenden.

Deinstalliere die Lösung

> **Note**
>
> Bereitstellungen, die über das Deployment-Dashboard erstellt wurden, sind nicht dafür vorgesehen, außerhalb der Lösung verwaltet zu werden. Stellen Sie sicher, dass Sie alle Bereitstellungen im Deployment-Dashboard löschen und bereinigen, bevor Sie den Stack darin löschen. CloudFormation

Sie können die Generative AI Application Builder on AWS-Lösung über die AWS-Managementkonsole oder über die AWS-Befehlszeilenschnittstelle deinstallieren. Sie müssen die Amazon S3-Buckets, Amazon Kendra Kendra-Indizes oder CloudWatch Logs, die mit dieser Lösung erstellt wurden, manuell löschen. AWS-Lösungen löschen Amazon S3 S3-Buckets, Amazon Kendra Kendra-Indizes oder CloudWatch Logs nicht automatisch, falls Sie Daten zur Aufbewahrung gespeichert haben.

Verwendung der AWS-Managementkonsole

1. Melden Sie sich bei der CloudFormation AWS-Konsole an.
2. Wählen Sie auf der Seite Stacks den Installations-Stack dieser Lösung aus.
3. Wählen Sie Löschen aus.

Verwenden der AWS-Befehlszeilenschnittstelle

Stellen Sie fest, ob die AWS-Befehlszeilenschnittstelle (AWS CLI) in Ihrer Umgebung verfügbar ist. Installationsanweisungen finden Sie unter Was ist die AWS-Befehlszeilenschnittstelle im AWS-CLI-Benutzerhandbuch. Nachdem Sie bestätigt haben, dass die AWS-CLI verfügbar ist, führen Sie den folgenden Befehl aus.

```
$ aws cloudformation delete-stack --stack-name <installation-stack-name>
```

Schritte zur manuellen Deinstallation

Löschen der Amazon S3 S3-Buckets

Diese Lösung ist so konfiguriert, dass der von der Lösung erstellte Amazon S3 S3-Bucket beibehalten wird, falls Sie den CloudFormation AWS-Stack löschen möchten, um einen versehentlichen Datenverlust zu verhindern. Nach der Deinstallation der Lösung können Sie diesen Amazon S3 S3-Bucket manuell löschen, wenn Sie die Daten nicht behalten müssen. Gehen Sie wie folgt vor, um den Amazon S3 S3-Bucket zu löschen.

1. Melden Sie sich bei der Amazon S3-Konsole an.
2. Wählen Sie im Navigationsbereich Buckets aus.
3. Suchen Sie die <stack-name>S3-Buckets.
4. Wählen Sie den S3-Bucket aus und wählen Sie Löschen.

Führen Sie den folgenden Befehl aus, um den S3-Bucket mithilfe der AWS-CLI zu löschen. Sie müssen den Bucket nicht zuerst leeren, wenn Sie die Option --force verwenden.

```
$ aws s3 rb s3://<bucket-name> --force
```

Löschen der Amazon Kendra Kendra-Indizes

Um versehentlichen Datenverlust zu verhindern, ist diese Lösung so konfiguriert, dass die von der Lösung erstellten Amazon Kendra Kendra-Indizes beibehalten werden, wenn der CloudFormation AWS-Stack gelöscht wurde. Nach der Deinstallation der Lösung können Sie die Amazon Kendra Kendra-Indizes, für die Sie keine Daten mehr aufbewahren müssen, manuell löschen. Gehen Sie wie folgt vor, um den Amazon Kendra Kendra-Index zu löschen.

1. Melden Sie sich bei der Amazon Kendra Kendra-Konsole an.
2. Wählen Sie im Navigationsbereich Indizes aus.
3. Suchen Sie den Index, den Sie löschen möchten, und wählen Sie ihn aus.
4. Wählen Sie Löschen aus, um den ausgewählten Index zu löschen.

Führen Sie den folgenden Befehl aus, um den Amazon Kendra Kendra-Index mithilfe der AWS-CLI zu löschen:

```
$ aws kendra delete-index --id<index-id>
```

Löschen der Protokolle CloudWatch

Um einen versehentlichen Datenverlust zu verhindern, haben wir diese Lösung so konfiguriert, dass die CloudWatch Protokolle aufbewahrt werden, falls Sie den CloudFormation Stack löschen. Nach der Deinstallation der Lösung können Sie die Protokolle manuell löschen, wenn Sie die Daten nicht behalten müssen. Gehen Sie wie folgt vor, um die CloudWatch Protokolle zu löschen.

1. Melden Sie sich bei der CloudWatch Amazon-Konsole an.
2. Wählen Sie im Navigationsbereich Protokollgruppen aus.
3. Suchen Sie nach den Protokollgruppen, die von der Lösung erstellt wurden.
4. Wählen Sie eine der Protokollgruppen aus.
5. Wählen Sie Actions und dann Delete aus.

Wiederholen Sie die Schritte, bis Sie alle Lösungsprotokollgruppen gelöscht haben.

Benutze die Lösung

Zugriff auf die Benutzeroberfläche

Während des Stack-Bereitstellungsprozesses (sowohl für das Deployment-Dashboard als auch für Anwendungsfälle) wird eine E-Mail an die konfigurierte E-Mail-Adresse gesendet. Die E-Mail enthält die temporären Anmeldeinformationen des Benutzers, mit denen er sich registrieren und auf die Weboberfläche zugreifen kann.

> **Note**
>
> Der DevOps Benutzer mit Zugriff auf die AWS-Managementkonsole muss dem Admin-Benutzer die CloudFront URL der Deployment-Dashboard-Benutzeroberfläche mitteilen, wenn der Stack abgeschlossen ist.

Für die Anwendungsfälle muss der Admin-Benutzer mit Zugriff auf die Benutzeroberfläche des Deployment-Dashboards dem Geschäftsbenutzer die CloudFront URL der Anwendungsfall-Benutzeroberfläche mitteilen, wenn die Bereitstellung abgeschlossen ist.

Sobald der Benutzer angemeldet ist, kann er mit der Lösung interagieren UIs, entweder mit dem Deployment-Dashboard im Fall von Administratoren oder mit dem Anwendungsfall im Fall von Geschäftsbenutzern.

Wie aktualisiert man ein Deployment

Wenn Sie sich auf der Startseite des Bereitstellungs-Dashboards (oder der Detailseite einer Bereitstellung) befinden, können Sie die von einer Bereitstellung verwendete Konfiguration bearbeiten. Sie können nur Bereitstellungen bearbeiten, die sich im Status CREATE_COMPLETE oder UPDATE_COMPLETE befinden.

Mit Ausnahme des Anwendungsfallnamens können alle anderen Optionen für eine Bereitstellung bearbeitet werden. Ändern Sie einfach die Werte, die Sie bearbeiten und erneut bereitstellen möchten.

Je nach Umfang der vorgenommenen Änderungen variiert die Dauer der erneuten Bereitstellung. Es kann einige Sekunden dauern, wenn sich einfache Einstellungen geändert haben (z. B.

Modellparameter), und mehr als 30 Minuten, wenn sich größere infrastrukturbezogene Optionen geändert haben (z. B. die Anforderung, den Amazon Kendra Kendra-Index für den Text-Anwendungsfall RAG zu erstellen).

Sobald die Bearbeitung erfolgreich abgeschlossen wurde, meldet der Anwendungsstatus den Status UPDATE_COMPLETE. Zu diesem Zeitpunkt können Sie über die CloudFront URL auf die bereitgestellte Benutzeroberfläche zugreifen und mit der geänderten Bereitstellung interagieren.

> **Note**
>
> Möglicherweise ist es einfacher, mehrere Bereitstellungen auszuführen, side-by-side wenn Sie verschiedene Einstellungen vergleichen möchten oder LLMs. Verwenden Sie die Clone-Funktion, um schnell eine bestehende Konfiguration zu verwenden, um eine neue Bereitstellung zu starten.

Wie klont man eine Bereitstellung

Wenn Sie auf der Startseite des Bereitstellungs-Dashboards (oder der Detailseite einer Bereitstellung) die von einer Bereitstellung verwendete Konfiguration klonen können. Beim Klonen einer Bereitstellung wird der Assistent zum Bereitstellen neuer Anwendungsfälle gestartet, wobei die meisten Felder jedoch bereits mit denselben Werten gefüllt sind.

Dies ist ein praktischer Vorgang, mit dem Sie schnell Bereitstellungen mit geänderten Einstellungen duplizieren, eine gelöschte Bereitstellung wiederbeleben oder mehrere Bereitstellungen LLMs in ansonsten identischen Bereitstellungen vergleichen können.

Wie lösche ich eine Bereitstellung

Wenn Sie sich auf der Startseite des Bereitstellungs-Dashboards (oder der Detailseite einer Bereitstellung) befinden, können Sie sie löschen, sobald Sie die Bereitstellung nicht mehr benötigen. Durch das Löschen einer Bereitstellung wird ein CloudFormation Stack-Löschvorgang aufgerufen und die Bereitstellung der Ressourcen für die Bereitstellung aufgehoben.

Standardmäßig verbleibt eine gelöschte Bereitstellung weiterhin auf dem Dashboard, um die Klonfunktion zu aktivieren. Um eine Bereitstellung vollständig aus dem Dashboard zu entfernen, sodass sie nicht mehr in der Benutzeroberfläche verfolgt wird, wählen Sie im Bestätigungsfenster für das Löschen die Option Dauerhaft löschen aus.

> **Important**
>
> Einige Ressourcen bleiben beim Löschen des Stacks zurück und müssen manuell gelöscht werden. Einzelheiten darüber, welche Ressourcen beibehalten werden und wie sie bereinigt werden, finden Sie im Abschnitt Manuelle Deinstallation.

Konfiguration eines Large Language Model (LLM)

Welches LLM für Ihren Anwendungsfall das Richtige ist, hängt von einer Vielzahl von Faktoren ab, die auf Ihre Bedürfnisse und die Art des Kundenerlebnisses zugeschnitten sind, das Sie kuratieren möchten. Diese Lösung scheint nicht präskriptiv zu sein, sondern zielt darauf ab, Ihnen die notwendigen Tools an die Hand zu geben, um zu beurteilen, was für Ihre Anwendung am besten geeignet ist.

Der KI-generierte Bereich entwickelt sich rasant. Daher liegt es an Ihnen, sich über die neuesten Modelle, Optimierungstechniken und Best Practices auf dem Laufenden zu halten, um sicherzustellen, dass Sie Ihren Kunden die richtigen Erlebnisse bieten.

> **Note**
>
> Wenn Sie mit nicht öffentlichen oder sensiblen Daten arbeiten, sollten Sie unbedingt eine LLM-Option mit AWS-Services (wie Amazon Bedrock oder Amazon SageMaker AI) auswählen. Dies verbessert die allgemeine Sicherheitslage Ihrer Bereitstellung, da die Daten in Ihrer Region und im AWS-Netzwerk gespeichert werden, verglichen mit der Verwendung eines LLM, das von einem Drittanbieter gehostet wird.

Verwendung von Amazon SageMaker AI als LLM-Anbieter

Ab Version 1.3.0 ist Amazon SageMaker AI als Modellanbieter für Text-Anwendungsfälle verfügbar. Mit dieser Funktion können Sie einen SageMaker AI-Inferenzendpunkt verwenden, der bereits im AWS-Konto in der Lösung vorhanden ist. Hier sind einige Möglichkeiten, um loszulegen.

> **Important**
>
> Die Lösung verwaltet nicht den Lebenszyklus Ihrer SageMaker KI-Endpunkte. Sie sind dafür verantwortlich, die SageMaker KI-Endpunkte zu löschen, sobald sie nicht mehr benötigt werden, damit keine zusätzlichen Kosten mehr anfallen.

Einen KI-Endpunkt erstellen SageMaker

Sie können Amazon SageMaker AI verwenden JumpStart, um schnell einen Endpunkt bereitzustellen.

Sie können auch einen SageMaker KI-Endpunkt verwenden, der auf Textgenerierung basiert, und die Bereitstellung mithilfe des SageMaker KI-Basisdienstes durchführen. In der SageMaker JumpStart KI-Dokumentation finden Sie eine schrittweise Anleitung zur Implementierung eines Inferenzmodells.

> **Note**
>
> models/LLMs Foundations sind in der Regel recht groß und erfordern häufig die Verwendung großer beschleunigter Recheninstanzen. Viele dieser größeren Instances sind möglicherweise nicht standardmäßig in Ihrem AWS-Konto verfügbar. Beachten Sie die standardmäßigen SageMaker KI-Kontingente und stellen Sie sicher, dass Sie vor der Bereitstellung eine Erhöhung des Kontingents beantragen, um häufige Bereitstellungsfehler zu vermeiden.

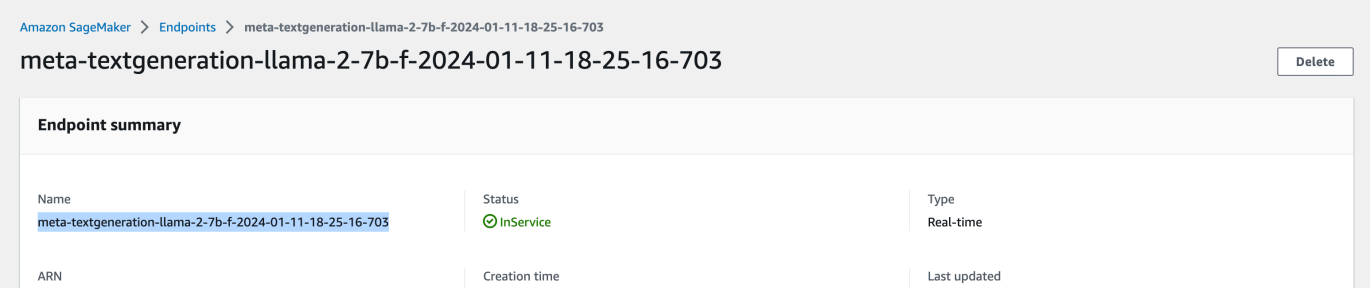
Verwenden Sie den SageMaker KI-Endpunkt, um eine Bereitstellung für einen Text-Anwendungsfall zu erstellen

So stellen Sie einen neuen Text-Anwendungsfall mithilfe eines SageMaker KI-Endpunkts für Inferenz bereit:

1. Erstellen Sie mit dem Assistenten für das Bereitstellungs-Dashboard einen neuen Anwendungsfall und füllen Sie die Formulare aus, bis Sie zur Seite mit der Modellauswahl gelangen.
2. Wählen Sie auf der Seite Modelle SageMaker AI als Modellanbieter aus. Dadurch wird ein benutzerdefiniertes Formular generiert, das drei wichtige Benutzereingaben erfordert:

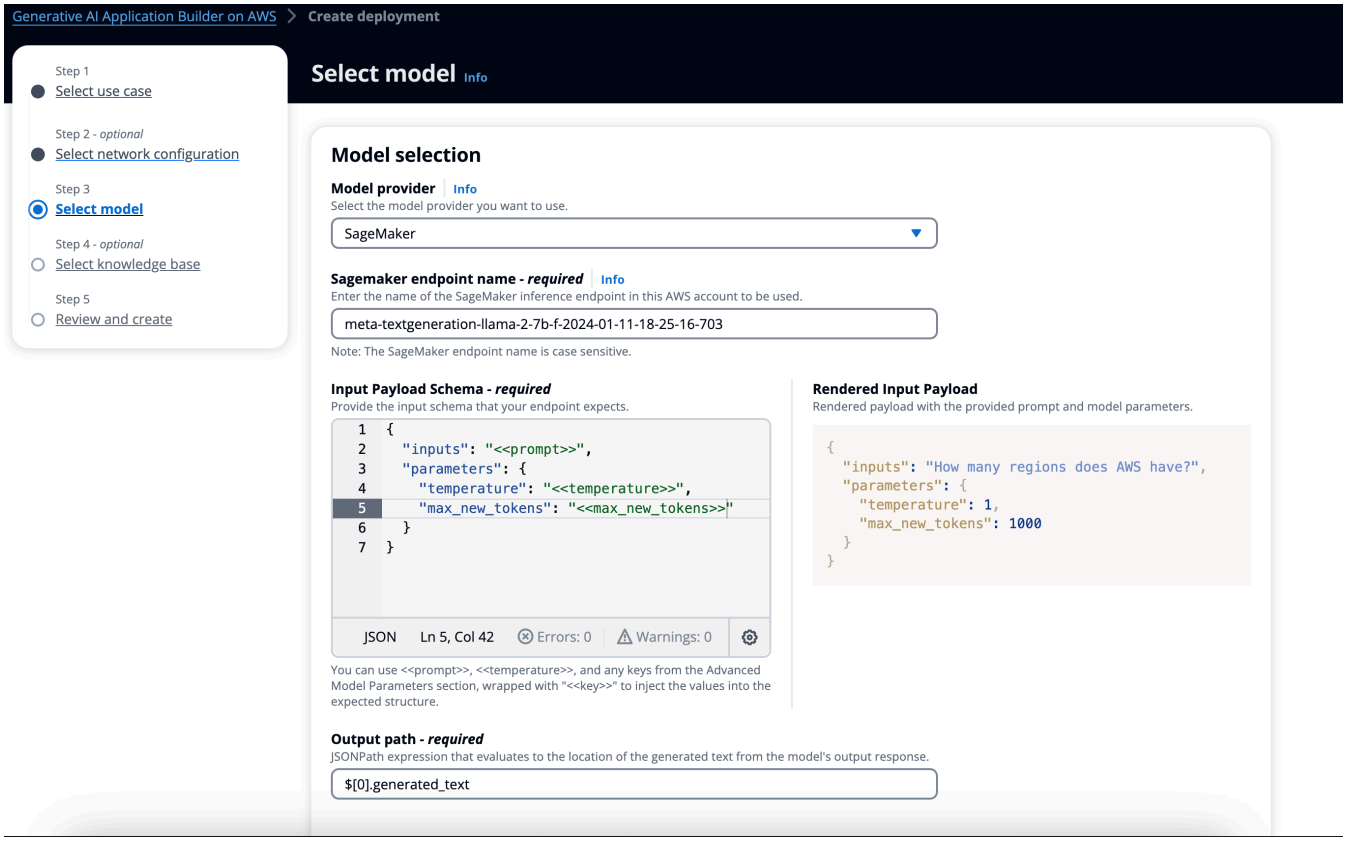
- Der Name des SageMaker KI-Endpunkts, den Sie verwenden möchten. DevOps Benutzer können dies über die AWS-Konsole abrufen. Beachten Sie, dass sich der Endpunkt in demselben Konto und derselben Region befinden muss, in der die Lösung bereitgestellt wird.

  Speicherort des Endpunktnamens auf der AWS-Konsole



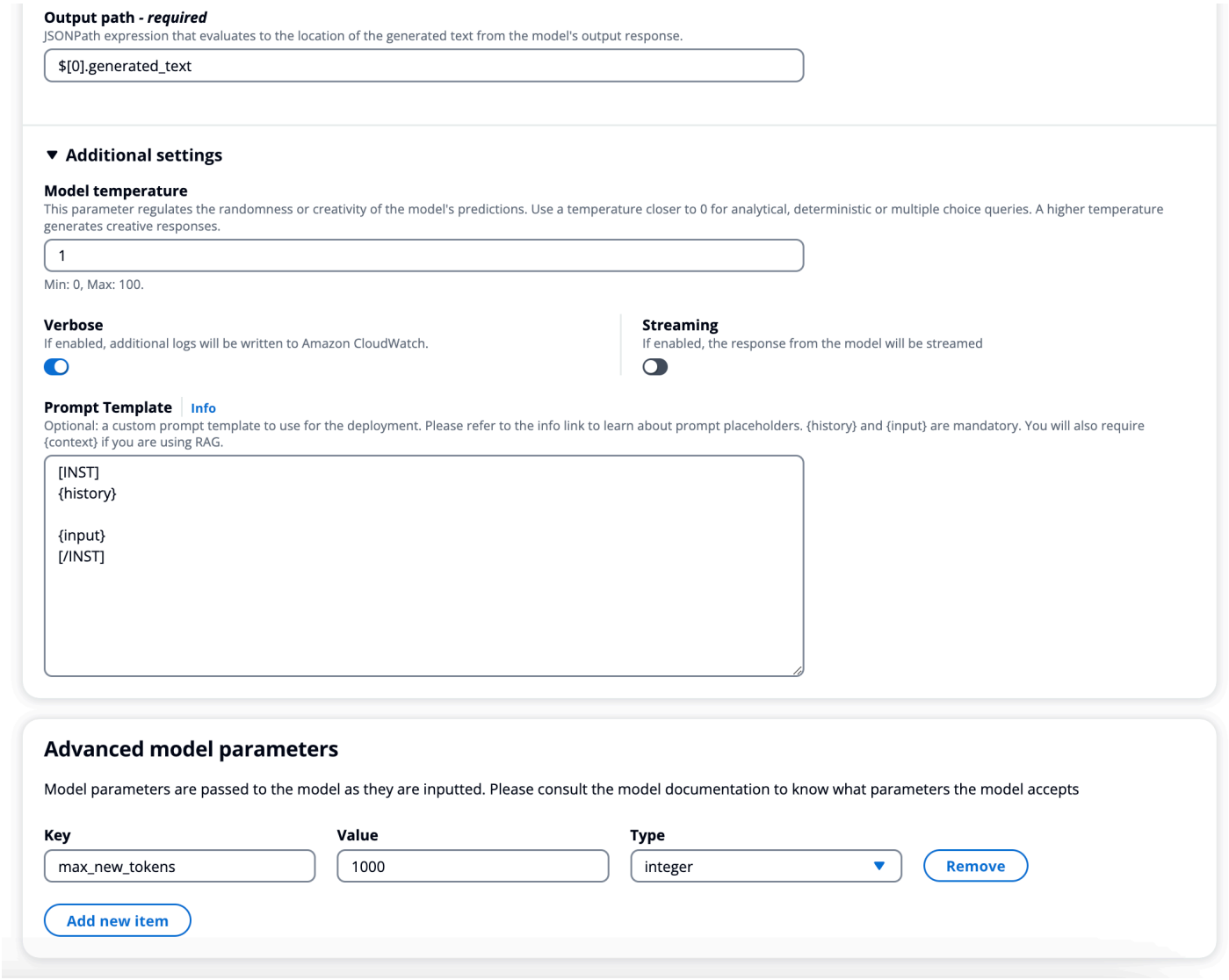
- Das Schema der Eingabe-Payload, die vom Endpunkt erwartet wird. Um die größtmögliche Anzahl von Endpunkten zu unterstützen, müssen Admin-Benutzer der Lösung mitteilen, wie ihr Endpunkt die Formatierung der Eingabe erwartet. Geben Sie im Assistenten zur Modellauswahl das JSON-Schema für die Lösung an, das an den Endpunkt gesendet werden soll. Sie können Platzhalter hinzufügen, um statische und dynamische Werte in die Payload der Anfrage einzufügen. Die folgenden Optionen sind verfügbar:
  - Obligatorische Platzhalter:\ < <prompt\ >\\ > werden dynamisch durch die vollständige Eingabe (z. B. Verlauf, Kontext und Benutzereingaben gemäß der Eingabeaufforderungsvorlage) ersetzt, die zur Laufzeit an den SageMaker KI-Endpunkt gesendet wird.
  - Optionale Platzhalter:\ <\ <temperature\ >\ > *,\ * sowie alle Parameter, die in erweiterten Modellparametern definiert sind, können dem Endpunkt zur Verfügung gestellt werden. Jede Zeichenfolge, die einen in\ <\ < and\ >\ > eingeschlossenen Platzhalter enthält (z. B. <max_new_tokens\ >\ <\\ >), wird durch den Wert des gleichnamigen erweiterten Modellparameters ersetzt.

    Beispiel für ein Eingabeschema — Festlegung der Pflichtfelder, der Eingabeaufforderung und der Temperatur sowie eines benutzerdefinierten erweiterten Parameters, max_new_tokens. Der Ausgabepfad muss als gültige Zeichenfolge angegeben werden JSONPath



3. Die Position der LLMs generierten Zeichenkettenantwort innerhalb der Ausgabe-Payload. Dies muss als JSONPath Ausdruck angegeben werden, um anzugeben, wo auf die endgültige Textantwort, die den Benutzern angezeigt wird, voraussichtlich innerhalb des Rückgabeobjekts und der Antwort des Endpunkts zugegriffen wird.

   Beispiel für das Hinzufügen erweiterter Modellparameter zur Verwendung innerhalb des SageMaker AI-Eingabeschemas (siehe Abbildung 2 für frühere Optionen/Einstellungen)



> **Note**
>
> SageMaker AI unterstützt jetzt das Hosten mehrerer Modelle hinter demselben Endpunkt. Dies ist die Standardkonfiguration bei der Bereitstellung eines Endpunkts in der aktuellen Version von SageMaker AI Studio (nicht Studio Classic).
> Wenn Ihr Endpunkt auf diese Weise konfiguriert ist, müssen Sie dem Abschnitt mit den erweiterten Modellparametern einen Wert hinzufügen InferenceComponentName, der dem Namen des Modells entspricht, das Sie verwenden möchten.

Erweiterte LLM-Einstellungen

Bei der Verwendung von Amazon Bedrock können Sie einige erweiterte Einstellungen für Ihre Modelle wie Amazon Bedrock Guardrails, Provisioned Throughput for Amazon Bedrock und zusätzliche Modellparameter konfigurieren.

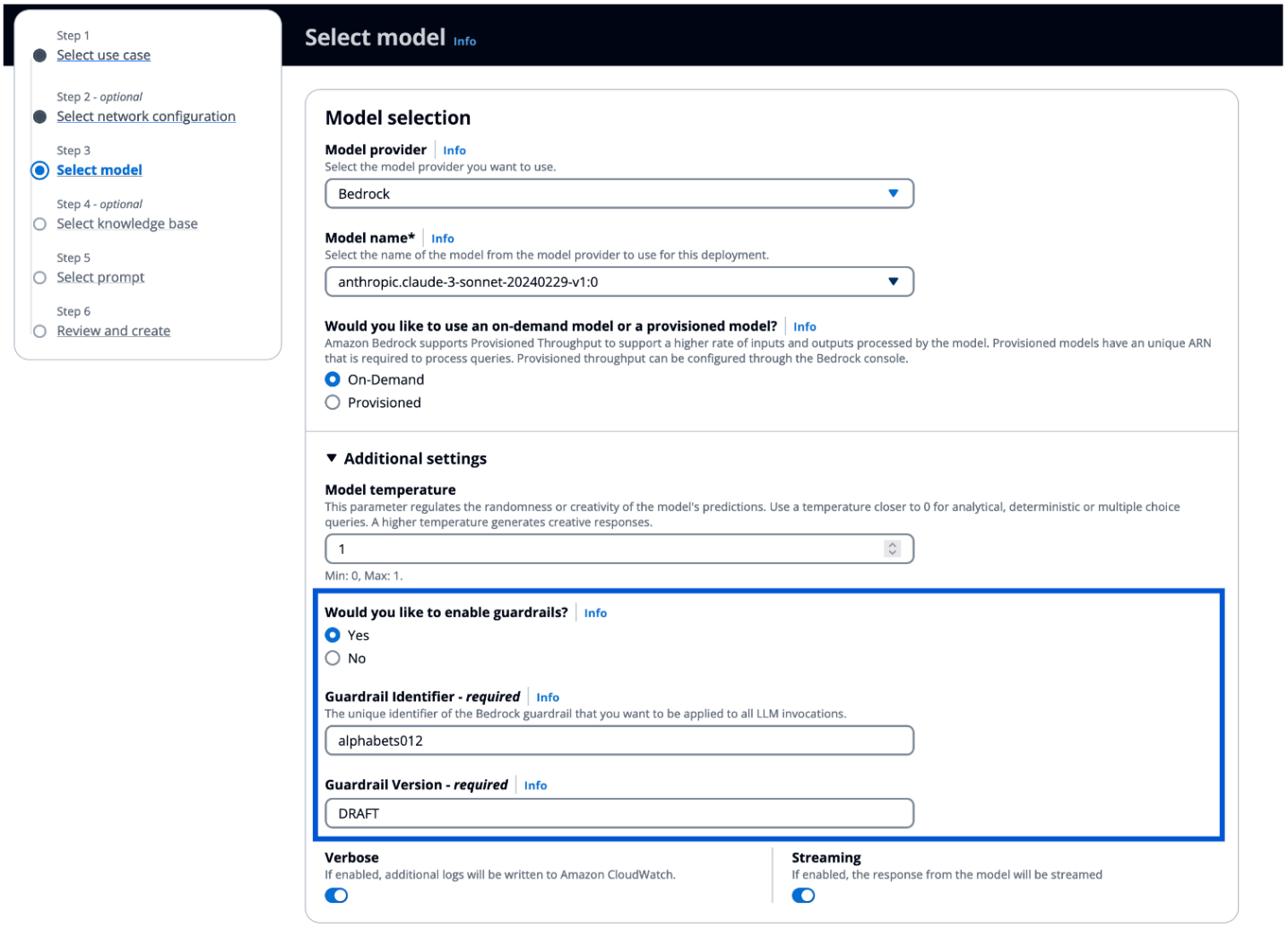
Integritätsschutz für Amazon Bedrock

Amazon Bedrock Guardrails ist eine Funktion von Amazon Bedrock, die Benutzereingaben und LLM-Antworten auf der Grundlage von vom Benutzer konfigurierten Richtlinien bewertet und eine zusätzliche Schutzebene bietet, unabhängig vom zugrunde liegenden LLM, das der Benutzer für einen Anwendungsfall auswählt. Ein Guardrail besteht aus zwei Richtlinien zur Vermeidung von Inhalten, die in unerwünschte oder schädliche Kategorien fallen:

1. Abgelehnte Themen, um eine Reihe von Themen zu definieren, die im Zusammenhang mit der Bewerbung des Benutzers unerwünscht sind, z. B. Anlageberatung in einer Finanzanwendung, und
2. Inhaltsfilter****ermöglichen das Filtern von Benutzereingaben oder das Modellieren von Antworten mit schädlichen Inhalten.

Für die Verwendung in der Generative AI Application Builder-Lösung muss ein Guardrail in der Amazon Bedrock-Konsole mithilfe des Assistenten Create Guardrail konfiguriert werden. Nach der Erstellung können Sie diese Guardrail zu Ihrem Chat-Anwendungsfall hinzufügen, der mit dem Generative AI Application Builder-Lösungsassistenten in den zusätzlichen Einstellungen im Schritt Modellauswahl erstellt wurde, indem Sie Ihre Guardrail-ID und Ihre Guardrail-Version angeben.

Zeigt den Bereitstellungsassistenten, der Amazon Bedrock Guardrails aktiviert



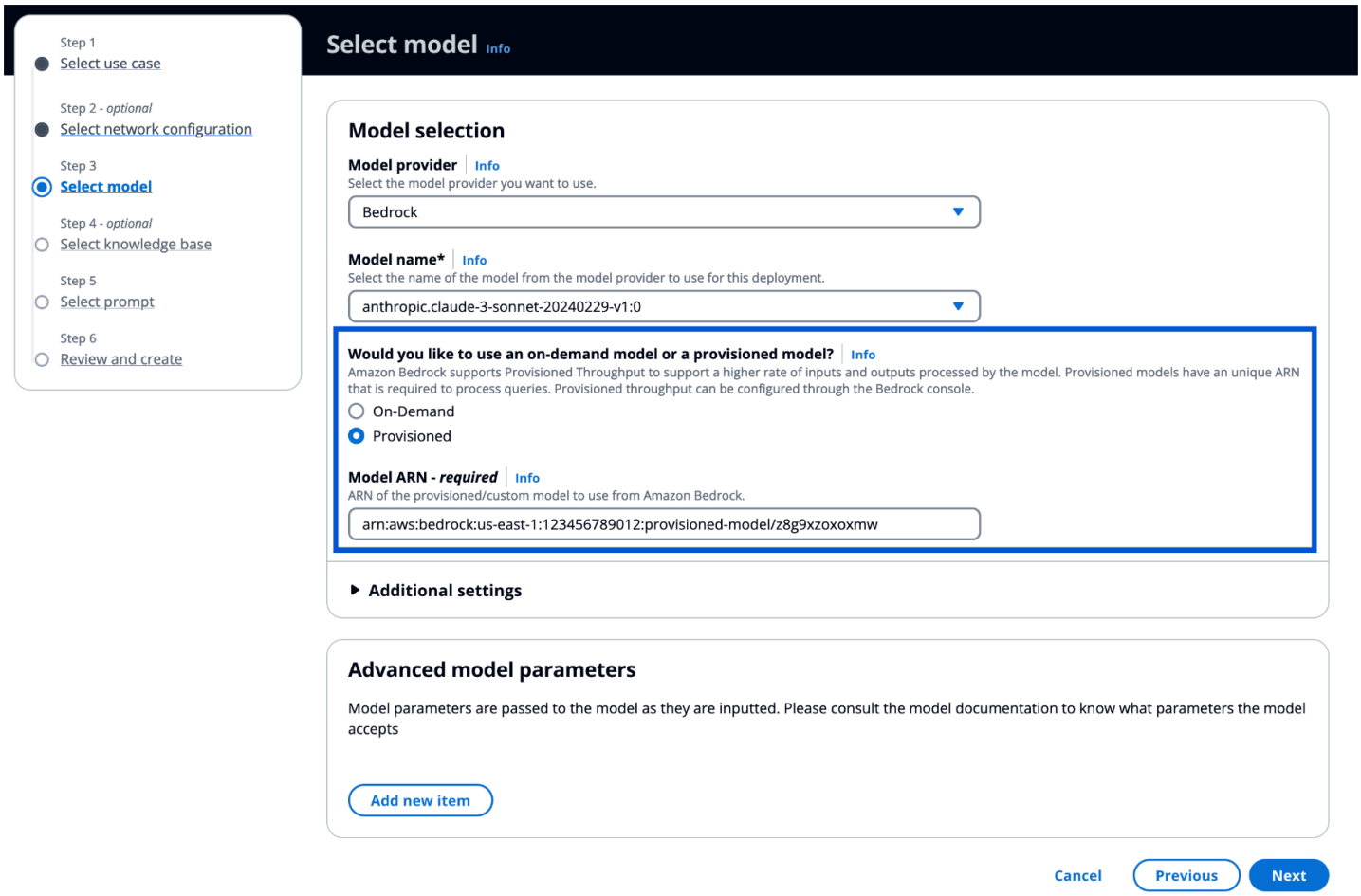
Bereitgestellter Durchsatz für Amazon Bedrock

Jedes On-Demand-Modell von Amazon Bedrock folgt einer regionsspezifischen Kontingentbegrenzung für Modellableitungen. Zum Beispiel ermöglicht Anthropic Claude 2.x auf Bedrock derzeit die Verarbeitung von 500 Anfragen und 500.000 Tokens pro Minute in den Regionen us-east-1 und us-west-2. Möglicherweise möchten Sie die Lösung auch mit Ihren fein abgestimmten oder bereits trainierten Modellen verwenden. Für solche Fälle ermöglicht Amazon Bedrock einen bereitgestellten Durchsatz, der die Ausführung großer konsistenter Inferenz-Workloads für Ihre Basis, fein abgestimmte oder fortlaufend vortrainierte Modelle für den Einsatz in produktionstauglichen Anwendungen ermöglicht.

Sobald Provisioned Throughput in der Amazon Bedrock-Konsole gekauft wurde, wird ein Modell-ARN zur Verwendung generiert. Sie können diesen Modell-ARN jetzt im Generative AI Application Builder-Assistenten im Schritt Modellauswahl angeben. Wählen Sie dazu Bedrock als Modellanbieter und

den Namen des Basismodells aus, das zur Generierung dieses bereitgestellten Modell-ARN in der Amazon Bedrock-Konsole verwendet wurde. Wählen Sie dann „Bereitgestelltes Modell“, wenn Sie zwischen On-Demand-Modellen und bereitgestellten Modellen wählen, und geben Sie Ihren Modell-ARN an.

Zeigt den Bereitstellungsassistenten — Aktivierung des bereitgestellten Durchsatzes für Amazon Bedrock



> **Note**
>
> Ihre Leitplanke und Ihr bereitgestellter Durchsatz müssen sich in derselben Region befinden wie das bereitgestellte Deployment Dashboard und die Anwendungsfall-Stacks.

Modellparameter

LLMs akzeptiert häufig eine Vielzahl von Parametern, die für die Implementierung spezifisch sind. Modellanbieter stellen häufig Unterlagen zur Verfügung, in denen der Satz der unterstützten Parameter und deren Verwendung beschrieben wird.

Die Lösung leitet die Modellparameter direkt an das zugrunde liegende Modell weiter. Daher ist es wichtig, sicherzustellen, dass die Parameter korrekt eingestellt sind. Aktuelle Informationen zu den unterstützten Parametern finden Sie in der Dokumentation des Modellanbieters.

Agent Builder konfigurieren

Agent Builder bietet umfassende Konfigurationsoptionen für die Erstellung produktionsreifer KI-Agenten. In diesem Abschnitt wird beschrieben, wie Agent Builder-Bereitstellungen konfiguriert und verwaltet werden.

Konfiguration der Systemaufforderung

Die Systemaufforderung definiert das Verhalten, die Persönlichkeit und die Fähigkeiten Ihres Agenten. So konfigurieren Sie die Systemaufforderung:

1. Navigieren Sie im Agent Builder-Assistenten zum Schritt Agent konfigurieren.
2. Bearbeiten Sie die Vorlage für die Systemaufforderung im Texteditor.
3. Fügen Sie klare Anweisungen hinzu für:
   - Rolle und Zweck des Agenten
   - Wie benutzt man die verfügbaren Tools (MCP-Server)
   - Einstellungen für die Formatierung von Antworten
   - Verhaltensrichtlinien
4. Verwenden Sie die Schaltfläche Auf Standard zurücksetzen, um bei Bedarf die ursprüngliche Vorlage wiederherzustellen.

Bewährte Methoden für Agentenaufforderungen:

- Machen Sie genaue Angaben zu den Fähigkeiten und Einschränkungen des Agenten
- Geben Sie klare Beispiele für das gewünschte Verhalten

- Fügen Sie Anweisungen zur Verwendung des Tools und zum Zeitpunkt des Aufrufs hinzu
- Definieren Sie die Erwartungen an das Antwortformat
- Legen Sie Grenzen für das Verhalten der Agenten fest

MCP-Serverintegration

MCP-Server (Model Context Protocol) bieten Agenten Zugriff auf Unternehmenstools und Datenquellen. So konfigurieren Sie MCP-Server:

1. Suchen Sie im Schritt „Agent konfigurieren“ den Abschnitt MCP-Server.
2. Wählen Sie im Dropdownmenü einen der verfügbaren MCP-Server aus.

> **Note**
>
> MCP-Server müssen vor der Bereitstellung des Agenten konfiguriert und zugänglich sein. Der Agent erkennt und verwendet automatisch Tools, die von den konfigurierten MCP-Servern bereitgestellt werden. Informationen zur Servereinrichtung und Toolkonfiguration finden Sie in der MCP-Dokumentation.

Speichereinstellungen

Agent Builder bietet zwei Speichertypen für die Verwaltung von Kontext und Wissen:

Kurzzeitgedächtnis

Standardmäßig für alle Agenten aktiviert:

- Behält den Konversationskontext innerhalb von Sitzungen bei
- Erfasst automatisch Benutzernachrichten und Agentenantworten
- Organisiert nach ActorID und sessionId für eine korrekte Isolierung
- Keine Konfiguration erforderlich

Langzeitgedächtnis

Optionale Funktion zum sitzungsübergreifenden Speichern von Erkenntnissen:

1. Suchen Sie im Schritt „Agent konfigurieren“ den Abschnitt Speicherkonfiguration.
2. Schalten Sie zur Aktivierung die Option Langzeitspeicher aktivieren um.
3. Wenn diese Option aktiviert ist, kann der Agent:
   - Wichtige Informationen aus Konversationen extrahieren und speichern
   - Rufen Sie den relevanten Kontext aus früheren Sitzungen ab
   - Bauen Sie Wissen über Benutzerpräferenzen und -historie auf

> **Note**
>
> Das AgentCore Langzeitgedächtnis verwendet Speicher mit semantischer Speicherstrategie und Standardeinstellungen für die Aufbewahrung.

Agent Builder-Bereitstellungen überwachen

Agent Builder bietet eine umfassende Überwachung mithilfe von CloudWatch Dashboards und Metriken.

Zugreifen auf Dashboards CloudWatch

1. Navigieren Sie in Ihrem AWS-Konto zur CloudWatch Konsole.
2. Wählen Sie in der linken Navigationsleiste Dashboards aus.
3. Suchen Sie das Dashboard mit dem Namen`AgentBuilder-<UseCaseId>`.
4. Sehen Sie sich Echtzeitmetriken und historische Leistungsdaten an.

Zugriff und Analyse von Protokollen

Agentenprotokolle sind unter CloudWatch Protokolle verfügbar:

1. Navigieren Sie in der AWS-Konsole zu CloudWatch Logs.
2. Suchen Sie nach Protokollgruppen mit dem `/aws/bedrock-agentcore/runtimes/` Präfix.
3. Verwenden Sie CloudWatch Insights, um Logs abzufragen und zu analysieren.
4. Suchen Sie nach bestimmten Anfrage IDs - oder Fehlermustern.

Workflow Builder konfigurieren

Workflow Builder ermöglicht die Orchestrierung mehrerer Agenten über einen Supervisor-Agenten, der die Arbeit an spezialisierte Agent Builder-Agenten delegiert.

Einen Workflow erstellen

1. Navigieren Sie zum Deployment Dashboard
2. Wählen Sie Workflow-Anwendungsfall erstellen aus
3. Konfigurieren Sie den Supervisor-Agenten:
   - Name: Beschreibender Name für den Workflow
   - Beschreibung: Zweck und Funktionen
   - Systemaufforderung: Anweisungen für die Delegierung und Koordination von Agenten
   - Modell: Basismodell für den Supervisor-Agenten

Bewährte Methoden für Eingabeaufforderungen durch Vorgesetzte:

- Beschreiben Sie klar, wann Sie die einzelnen Spezialagenten einsetzen sollten
- Fügen Sie Anweisungen zum Aggregieren der Ergebnisse mehrerer Agenten hinzu
- Definieren Sie Erwartungen an die Formatierung der Antworten
- Legen Sie Grenzen für das Delegationsverhalten fest

Auswahl des Agenten

Wählen Sie Agent Builder-Agenten aus, die als spezialisierte Agenten aufgenommen werden sollen:

1. Klicken Sie in der Workflow-Konfiguration auf Agent hinzufügen
2. Suchen oder suchen Sie nach verfügbaren Agent Builder-Agenten
3. Überprüfen Sie die Agentenbeschreibungen
4. Wählen Sie Agenten aus, die in den Workflow aufgenommen werden sollen

Beschreibungen der Agenten

Der Supervisor-Agent entscheidet anhand der Agentenbeschreibungen, an welchen Agenten er delegieren soll. Stellen Sie sicher, dass die Beschreibungen Folgendes klar erläutern:

- Spezialgebiet oder Fähigkeit des Agenten
- Arten von Aufgaben, die der Agent bearbeitet
- Input-/Output-Erwartungen

Workflows testen

Nach der Bereitstellung:

1. Greifen Sie über das Deployment Dashboard auf den Workflow zu
2. Testen Sie mit Abfragen, für die mehrere Agenten erforderlich sind
3. Überwachen Sie die Agentendelegierung in CloudWatch Protokollen
4. Überprüfen Sie die Qualität der Antworten und die Delegationsmuster
5. Passen Sie die Aufforderung des Vorgesetzten an, falls die Delegierung nicht optimal ist

Tipps zur Verwaltung der Limits für Modell-Tokens

Hinweis: Die Lösung versucht nicht direkt, die durch verschiedene Faktoren auferlegten Token-Limits zu verwalten LLMs. Testen Sie und stellen Sie sicher, dass Ihre Eingabeaufforderung innerhalb der vom Modellanbieter festgelegten verfügbaren Grenzwerte bleibt.

Versuchen Sie Folgendes, um die Größe der Eingabeaufforderungen zu kontrollieren:

1. Machen Sie sich mit den Einschränkungen vertraut, die das Modell, das Sie verwenden möchten, auferlegt. Diese Werte können sich je nach Modell erheblich unterscheiden. Daher ist es wichtig, dass Sie wissen, wie hoch Ihr verfügbares Budget ist, bevor Sie beginnen.
2. Denken Sie bei Ihrer ersten Aufforderung an dieses Budget und überlegen Sie, wie viel Sie für dynamische Elemente der Aufforderung sparen möchten. Zum Beispiel Benutzereingaben, Chat-Verlauf, Dokumentauszüge usw.
3. Legen Sie auf der Seite zur Konfiguration der Eingabeaufforderung ein Limit für die Größe des nachfolgenden Verlaufs fest, um die Anzahl der Konversationsrunden zu begrenzen, die in der Aufforderung enthalten sind.
4. Legen Sie im Konfigurationsassistenten der Knowledge Base Beschränkungen für die Rückgabe von Dokumenten fest. Sie müssen versuchen, das richtige Gleichgewicht zwischen der Bereitstellung von ausreichend Kontext für das LLM zur Ausführung der Aufgabe zu finden, aber nicht so sehr, dass die Token-Limits überschritten oder die Latenz negativ beeinflusst wird.

5. Lassen Sie etwas Puffer übrig. Planen Sie nicht für den typischen Fall ein, sondern denken Sie über Randfälle wie lange Eingabeabfragen, umfangreiche Dokumentauszüge oder lange Konversationen nach und experimentieren Sie mit ihnen.

Schritte zum Erstellen eines MCP-Server-Docker-Images

Um MCP-Server (Model Context Protocol) mit Generative AI Application Builder auf AWS zu verwenden, benötigen Sie als ersten Schritt ein Docker-Image, das in einem privaten Amazon ECR-Repository erstellt und gespeichert wurde.

> **Note**
>
> Derzeit können bestehende bereitgestellte MCP-Server in Amazon Bedrock AgentCore Runtime nicht nach GAAB exportiert werden. Damit MCP-Server an Agenten angehängt werden können, die über GAAB erstellt wurden, müssen sie über GAAB erstellt werden.

Schritt 1: Erstellen Sie Ihren MCP-Server

Zunächst müssen Sie Ihre MCP-Serverimplementierung fertig haben. Detaillierte Anweisungen zur Erstellung eines MCP-Servers finden Sie im Amazon Bedrock AgentCore Developer Guide — Create an MCP server.

Wir empfehlen die folgende Projektstruktur:

```
.
### __init__.py
### extras/
#   ### extra_dependencies.py
#   ### Dockerfile
### requirements.txt
### server.py <-- Server Entry point
```

Für die Dockerfile-Struktur empfehlen wir die Verwendung eines Formats, das dem folgenden Beispiel ähnelt:

```
FROM ghcr.io/astral-sh/uv:python3.13-bookworm-slim
WORKDIR /app
```

```
# All environment variables in one layer
ENV UV_SYSTEM_PYTHON=1 \
    UV_COMPILE_BYTECODE=1 \
    UV_NO_PROGRESS=1 \
    PYTHONUNBUFFERED=1 \
    DOCKER_CONTAINER=1 \
    AWS_REGION=us-east-1 \
    AWS_DEFAULT_REGION=us-east-1

COPY requirements.txt requirements.txt
# Install from requirements file
RUN uv pip install -r requirements.txt

RUN uv pip install aws-opentelemetry-distro>=0.10.1

# Signal that this is running in Docker for host binding logic
ENV DOCKER_CONTAINER=1

# Create non-root user
RUN useradd -m -u 1000 bedrock_agentcore
USER bedrock_agentcore

EXPOSE 9000
EXPOSE 8000
EXPOSE 8080

# Copy entire project (respecting .dockerignore)
COPY . .

# Use the full module path
CMD ["opentelemetry-instrument", "python", "-m", "server"]
```

Schritt 2: Testen Sie Ihren MCP-Server lokal

Vor der Bereitstellung auf AWS ist es wichtig, Ihren MCP-Server lokal zu testen, um sicherzustellen, dass er erwartungsgemäß funktioniert. Detaillierte Anweisungen zu lokalen Tests finden Sie im Amazon Bedrock AgentCore Developer Guide — Testen Sie Ihren MCP-Server lokal.

Schritt 3: Auf Amazon ECR bereitstellen

Sobald Ihr MCP-Server lokal erstellt und getestet wurde, gehen Sie wie folgt vor, um ihn auf Amazon ECR bereitzustellen:

1. Stellen Sie sicher, dass Sie die neueste Version von AWS CLI und Docker installiert haben. Weitere Informationen finden Sie unter Erste Schritte mit Amazon ECR.
2. Rufen Sie ein Authentifizierungstoken ab und authentifizieren Sie Ihren Docker-Client bei Ihrer Registrierung. Verwenden Sie die AWS-CLI:

```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --
password-stdin <account-id>.dkr.ecr.us-east-1.amazonaws.com
```

3. Erstellen Sie Ihr Docker-Image mit dem folgenden Befehl. Informationen zum Erstellen einer Docker-Datei von Grund auf finden Sie in der Docker-Dokumentation. Sie können diesen Schritt überspringen, wenn Ihr Image bereits erstellt wurde:

```
docker build -t <repository-name> .
```

4. Nachdem der Build abgeschlossen ist, taggen Sie Ihr Image, damit Sie das Image in dieses Repository übertragen können:

```
docker tag <repository-name>:latest <account-id>.dkr.ecr.us-east-1.amazonaws.com/
<repository-name>:latest
```

5. Führen Sie den folgenden Befehl aus, um dieses Image in Ihr neu erstelltes AWS-Repository zu übertragen:

```
docker push <account-id>.dkr.ecr.us-east-1.amazonaws.com/<repository-name>:latest
```

Vollständige Anweisungen zur Bereitstellung finden Sie im Amazon Bedrock AgentCore Developer Guide — Deploy your MCP server to AWS.

Schritt 4: Verwenden Sie die ECR-URI in GAAB

Nachdem Sie Ihr Docker-Image erfolgreich auf Amazon ECR übertragen haben, kopieren Sie den Image-URI aus der ECR-Konsole. Sie verwenden diesen URI, wenn Sie Ihren MCP-Server über den Bereitstellungsassistenten von Generative AI Application Builder on AWS bereitstellen.

Schritte zum Erstellen verschiedener MCP Gateway-Ziele

Mit Amazon Bedrock AgentCore Gateway können Sie bestehende AWS-Services APIs in MCP-Tools umwandeln, die von Ihren Agenten verwendet werden können. Das Gateway unterstützt mehrere Zieltypen, sodass Sie verschiedene Backend-Services nahtlos integrieren können.

Die folgenden Zieltypen werden unterstützt:

- Lambda-Ziele: Verwandeln Sie AWS Lambda Lambda-Funktionen in MCP-Tools. Eine ausführliche Anleitung finden Sie im Amazon Bedrock AgentCore Developer Guide — Add Lambda targets.
- OpenAPI-Ziele: Verwenden Sie OpenAPI-Spezifikationen, um REST APIs als MCP-Tools zu definieren und verfügbar zu machen. Eine ausführliche Anleitung finden Sie im Amazon Bedrock AgentCore Developer Guide — OpenAPI-Schema.
- Ziele von Smithy: Erstellen Sie MCP-Tools mithilfe von Smithy-Modelldefinitionen für typsichere API-Integrationen. Eine ausführliche Anleitung finden Sie im Amazon Bedrock AgentCore Developer Guide — Building Smithy targets.
- MCP-Serverziele: Stellen Sie über URL-Endpunkte eine direkte Verbindung zu externen MCP-Servern her, sodass Sie vorhandene MCP-Server integrieren können. Eine ausführliche Anleitung finden Sie im Amazon Bedrock AgentCore Developer Guide — MCP-Serverziele.

Weitere Beispiele und Tutorials zur Erstellung von MCP Gateway-Zielen finden Sie im Amazon Bedrock AgentCore Samples Repository.

Konfiguration einer Wissensdatenbank

In diesem Abschnitt wird beschrieben, wie Sie Daten in die Wissensdatenbank aufnehmen, die Sie für die Lösung ausgewählt haben. Die Lösung unterstützt derzeit Amazon Kendra und Amazon Bedrock Knowledge Bases als Wissensdatenbanken für Ihre RAG-basierte Anwendungsfallbereitstellung.

Amazon Kendra

Wenn Sie Amazon Kendra als Wissensdatenbank verwenden, finden Sie im Amazon Kendra Developer Guide Informationen darüber, wie Sie verschiedene Datenquellen-Konnektoren verwenden können, um Daten aus einer Vielzahl von Quellen aufzunehmen.

Wichtig: Um versehentlichen Datenverlust zu verhindern, löscht die Lösung den Kendra-Index (unabhängig davon, ob er von der Lösung erstellt wurde oder nicht) nicht automatisch, wenn ein

Deployment oder ein Stack gelöscht wird. Wenn Sie Ihre Wissensdatenbank löschen und keine Kosten mehr anfallen möchten, finden Sie im Abschnitt Manuelle Deinstallation weitere Informationen darüber, welche Ressourcen beibehalten werden und wie Sie sie bereinigen können.

Amazon Bedrock Wissensdatenbanken

Amazon Bedrock Knowledge Bases können durch eine Vielzahl verschiedener Vector Stores unterstützt werden, von denen jeder die Fähigkeit besitzt, Ihre Daten zu indizieren. Informationen zum Einrichten und Befüllen Ihrer Wissensdatenbank finden Sie im Amazon Bedrock-Benutzerhandbuch. Insbesondere sollten Sie:

- Richten Sie zuerst Ihre Datenquelle ein
- Richten Sie dann einen Vektorindex für Ihre Wissensdatenbank in einem unterstützten Vektorspeicher ein. Beachten Sie, dass dies übersprungen werden kann, wenn Sie bei der Erstellung der Wissensdatenbank die Option „Schnell einen neuen Vektorspeicher erstellen“ in der Bedrock-Konsole verwenden.
- Schließlich können Sie die Wissensdatenbank erstellen und Ihre konfigurierten Datenquellen synchronisieren.

Erweiterte Einstellungen für die Wissensdatenbank

Erweiterte Knowledge Base-Einstellungen wie Knowledge Base-Filterung und RAG mit rollenbasierter Zugriffskontrolle sind für die Verwendung mit der Lösung verfügbar. Die Wissensdatenbank-Filterung kann auf jede der Knowledge Bases angewendet werden, während RAG mit Role Based Access Control speziell für Amazon Kendra verfügbar ist.

Filterung der Wissensdatenbank

Mit der Lösung können Sie Amazon Kendra-Attributfilter oder Bedrock-Abruffilter für die Wissensdatenbank angeben, wenn Sie einen Anwendungsfall im Abschnitt Erweiterte RAG-Konfigurationen des Wissensdatenbankschritts des Assistenten bereitstellen. Diese Filter definieren, wie Datenquellen in der Wissensdatenbank abgefragt werden, z. B. Suchstrategien, Sprachen des zugrunde liegenden Dokuments, bei dem es sich um Abfragen handelt, usw.

In beiden Fällen wird ein JSON-Objekt verwendet, um die Filtereinstellungen gemäß dem Format anzugeben, das in der jeweiligen Servicedokumentation (wie oben verlinkt) angegeben ist.

Beispiel 1: Kendra AttributeFilter

```
{
 "EqualsTo": {
 "Key": "_language_code",
 "Value": {
 "StringValue": "es"
 }
 }
}
```

Beispiel 2: Grundgestein RetrievalFilter

```
{
 "equals": {
 "key": "language",
 "value": "es"
 }
}
```

RAG mit rollenbasierter Zugriffskontrolle mit Amazon Kendra

Mit der rollenbasierten Zugriffskontrolle (RBAC) können Sie steuern, welche Benutzer oder Gruppen auf bestimmte Dokumente in Ihrem Amazon Kendra Kendra-Index zugreifen oder bestimmte Dokumente in ihren Suchergebnissen sehen können. Gehen Sie wie folgt vor, um RBAC für Ihre Amazon Kendra Index-ID mit Ihrem Anwendungsfall Generative AI Application Builder on AWS (GAAB) zu konfigurieren:

1. Amazon Kendra Index konfigurieren

1. Stellen Sie sicher, dass Sie Amazon Kendra Kendra-Index erstellt und ihm mindestens eine Datenquelle hinzugefügt haben.
2. Konfigurieren Sie die Zugriffskontrolle für Ihre Datenquelle auf der Grundlage von Benutzergruppen. Folgen Sie für eine S3-Datenquelle den Anweisungen in der Dokumentation, um Zugriffskontrolllisten (ACLs) mit denselben Gruppennamen einzurichten, die in Ihrem Amazon Cognito Cognito-Benutzerpool erstellt wurden. Dadurch wird sichergestellt, dass Benutzer nur auf die Dokumente und Suchergebnisse zugreifen können, zu deren Anzeige sie aufgrund ihrer Gruppenmitgliedschaft berechtigt sind.

> **Note**
>
> Belassen Sie im Kendra-Index, den Sie erstellt haben, unter Benutzerzugriffskontrolle den Wert Nein für Token-basierte Benutzerzugriffskontrolle. Wenn Sie die rollenbasierte Zugriffskontrolle in Schritt 2 aktivieren, extrahiert Generative AI Application Builder auf AWS die entsprechenden Ansprüche aus dem Benutzerauthentifizierungstoken und erstellt einen Attributfilter.

2. Stellen Sie den RAG-Anwendungsfall mit dem GAAB Deployment Wizard bereit

1. Folgen Sie den Anweisungen des Assistenten auf dem Bildschirm im GAAB Deployment Wizard, bis Sie Schritt 4 des Assistenten zur Konfiguration von RAG erreichen.
2. Wählen Sie im Schritt Wissensdatenbank auswählen des Bereitstellungsassistenten Amazon Kendra als Wissensdatenbanktyp aus.
3. Geben Sie an, ob Sie über einen Amazon Kendra Kendra-Index verfügen oder ob Sie einen neuen erstellen möchten. Wenn Sie über einen vorhandenen Index verfügen, geben Sie die ID Ihres Amazon Kendra-Indexes an, der mit auf Benutzergruppen basierenden Zugriffskontrolllisten (ACLs) konfiguriert wurde.
4. Aktivieren Sie die Option Rollenbasierte Zugriffskontrolle. Diese Option stellt sicher, dass die vom Amazon Kendra Kendra-Index zurückgegebenen Suchergebnisse auf der Grundlage der Rollen- und Gruppenberechtigungen des Benutzers gefiltert werden.
5. Überprüfen Sie den Anwendungsfall und stellen Sie ihn bereit.

3. Amazon Cognito konfigurieren

1. Suchen Sie den Amazon Cognito Cognito-Benutzerpool, der von Ihrer GAAB-Bereitstellung verwendet wird. Dieser Amazon Cognito Cognito-Benutzerpool wird normalerweise vom CloudFormation Hauptstapel des Bereitstellungs-Dashboards erstellt.
2. Erstellen Sie neue Benutzer im Amazon Cognito Cognito-Benutzerpool. Wählen Sie beim Erstellen von Benutzern die Option „E-Mail-Einladung senden“, damit Benutzer temporäre Anmeldeinformationen per E-Mail erhalten. Dadurch können sich neue Benutzer registrieren und auf die GAAB-Anwendung zugreifen.
3. Erstellen Sie Benutzergruppen im Amazon Cognito Cognito-Benutzerpool. Stellen Sie sicher, dass die Gruppennamen genau mit den in Ihrem Amazon Kendra Kendra-Index ACLs

konfigurierten Gruppen übereinstimmen. Dies ist entscheidend für die Aktivierung von RBAC, da die Gruppenmitgliedschaft des Benutzers bestimmt, auf welche Suchergebnisse er zugreifen kann.

4. Ordnen Sie Benutzer auf der Grundlage ihrer Rollen und Zugriffsberechtigungen den entsprechenden Gruppen zu. Benutzer müssen sowohl zu der Gruppe hinzugefügt werden, die für die Amazon Kendra-Index-ACL erforderlich ist, als auch zu der anwendungsfallspezifischen Gruppe, die während der GAAB-Bereitstellung erstellt wurde. Dadurch wird sichergestellt, dass Benutzer über die erforderlichen Berechtigungen für den Zugriff auf den jeweiligen Anwendungsfall und die entsprechenden Suchergebnisse verfügen.

Wenn Sie diese Schritte ausführen, haben Sie die rollenbasierte Zugriffskontrolle (RBAC) für Ihre GAAB-Bereitstellung konfiguriert. Dadurch wird sichergestellt, dass Benutzer nur auf die Informationen und Funktionen zugreifen und mit ihnen interagieren können, für die sie auf der Grundlage der ihnen zugewiesenen Benutzergruppe und den ihnen zugewiesenen Berechtigungen autorisiert sind.

> **Note**
>
> Derzeit unterstützt nur Amazon Kendra RBAC für Wissensdatenbanken im Generative AI Application Builder auf AWS. Für Amazon Bedrock Knowledge Base wird RBAC nicht unterstützt, aber Sie können Metadatenfilter verwenden, um ein gewisses Maß an Filterung zu erreichen. Weitere Informationen finden Sie im Amazon Bedrock-Benutzerhandbuch.

Konfiguration Ihrer Eingabeaufforderungen

Der Assistent für das Bereitstellungs-Dashboard verfügt über einen Schritt zur Konfiguration von Eingabeaufforderungen, mit dem Sie die Benutzeroberfläche und die Vorlage für die Interaktionen zwischen Benutzern und dem KI-Modell anpassen können. Die korrekte Konfiguration dieser Einstellungen ist entscheidend, um genaue und relevante Antworten vom KI-Assistenten zu erhalten.

In diesem Abschnitt werden das Gesamterlebnis und das Verhalten der KI-Eingabeaufforderung gesteuert.

- Max. Länge der Prompt-Vorlage: Diese Einstellung bestimmt die maximale Länge (in Zeichen) der Prompt-Vorlage. Ein höherer Wert ermöglicht es, dem KI-Modell mehr Kontext zur Verfügung zu stellen, was möglicherweise zu genaueren Antworten führt. Zu lange Eingabeaufforderungen können jedoch auch zu Geräuschen führen und sich negativ auf die Leistung auswirken. Für

Amazon Bedrock-Modelle werden die Standardwerte für die maximale Länge der Prompt-Vorlage (in Zeichen) anhand der zugrunde liegenden Modell-Token-Grenzwerte berechnet. Wenn Sie einen Modellnamen in Bedrock bearbeiten und ändern, ist die Schaltfläche „Auf Standard zurücksetzen“ markiert und kann verwendet werden, um die Standardeinstellungen des neu ausgewählten Modells zu übernehmen. Für Amazon SageMaker AI-Modelle werden angemessene Standardwerte bereitgestellt. Es wird jedoch empfohlen, dass Sie Ihr zugrundeliegendes Modell überprüfen und diese maximale Länge der Eingabeaufforderungsvorlagen und die Länge des eingegebenen Texts entsprechend wählen. Weitere Informationen finden Sie im Abschnitt Tipps zur Verwaltung von Modell-Token-Limits.

- Maximale Länge des Eingabetextes: Diese Einstellung begrenzt die maximale Länge (in Zeichen) des Eingabetextes des Benutzers. Längere Eingaben können irrelevante Informationen enthalten, was das Risiko erhöht, irrelevante oder ungenaue Antworten aus dem KI-Modell zu erhalten.
- Bearbeitung von Benutzeraufforderungen: Mit dieser Option können Sie die Möglichkeit aktivieren oder deaktivieren, dass Benutzer die Vorlage für die Aufforderung über die Chat-Benutzeroberfläche ändern können. Die Deaktivierung dieser Funktion kann dazu beitragen, die Konsistenz aufrechtzuerhalten und unbeabsichtigte Änderungen an der Eingabeaufforderung zu verhindern.

Vorlage für eine Aufforderung

In diesem Abschnitt können Sie die tatsächliche Vorlage für die Aufforderung definieren, die vom KI-Modell verwendet wird. Die Vorlage für die Aufforderung folgt in der Regel einer Struktur, die Platzhalter für verschiedene Komponenten enthält, z. B. für Benutzereingaben, Referenzpassagen und den Chat-Verlauf.

- Vorlage für Eingabeaufforderung: Dies ist der Haupttextbereich, in den Sie die gewünschte Eingabeaufforderungsvorlage schreiben oder einfügen können. Die Vorlage sollte so gestaltet sein, dass sie den erforderlichen Kontext und die erforderlichen Anweisungen für das KI-Modell bietet. Sie enthält in der Regel die folgenden Platzhalter:
  - `{input}`: Dieser Platzhalter ist für KI-Bereitstellungen von Sagemaker obligatorisch und wird durch die Eingabe oder Abfrage des Benutzers ersetzt.
  - `{history}`: Dieser Platzhalter ist für KI-Bereitstellungen von Sagemaker obligatorisch und wird durch den Chat-Verlauf der aktuellen Konversation ersetzt.
  - `{context}`: Dieser Platzhalter ist für RAG-Bereitstellungen obligatorisch und wird durch die Dokumentauszüge aus der konfigurierten Wissensdatenbank ersetzt.

- Frage umformulieren? : Diese Option (nur für RAG-Bereitstellungen verfügbar) bestimmt, ob die ursprüngliche Eingabeabfrage des Benutzers umformuliert oder eindeutig formuliert werden soll, bevor sie an das KI-Modell übergeben wird. Eine Umformulierung der Abfrage kann dem Modell manchmal helfen, die Absicht des Benutzers besser zu verstehen, was möglicherweise zu genaueren Antworten führt.

Bei der Konfiguration der Vorlage und der Benutzeroberfläche für die Aufforderung ist es wichtig, ein Gleichgewicht zwischen der Bereitstellung von ausreichend Kontext und Anweisungen für das KI-Modell zu finden und gleichzeitig zu lange oder irrelevante Informationen zu vermeiden, die zu Störungen oder Leistungseinbußen führen können.

Erweiterte Einstellungen für die Aufforderung

In diesem Abschnitt können Sie steuern, wie der Konversationsverlauf dem KI-Modell präsentiert wird.

- Größe des nachfolgenden Verlaufs: Diese Einstellung bestimmt die Anzahl der vorherigen Nachrichten, die in der endgültigen Eingabeaufforderung enthalten sein sollen. Wenn dieser Wert auf Null gesetzt wird, würde weder in die Eingabeaufforderungsvorlage noch in die Vorlage für die Disambiguierungsaufforderung ein Verlauf eingefügt. Bitte beachten Sie: Auch wenn dieser Wert auf Null gesetzt ist, muss in den Vorlagen für die Eingabeaufforderung immer noch ein Platzhalter für {history} vorhanden sein. Zur Laufzeit wird er durch eine leere Zeichenfolge ersetzt.
  - Hinweis: Es wird empfohlen, für diesen Wert eine gerade Zahl anzugeben. Die Angabe einer ungeraden Zahl würde dazu führen, dass nur die KI-Antwort einer gepaarten Interaktion zurückgegeben würde.
- Menschliches Präfix: Dies ist das Präfix, das zur Identifizierung von Nachrichten verwendet wird, die vom Benutzer im Konversationsverlauf gesendet wurden.
- KI-Präfix: Dies ist das Präfix, das zur Identifizierung von Nachrichten verwendet wird, die vom KI-Modell im Konversationsverlauf zurückgegeben wurden.

Konfiguration der Eingabeaufforderung zur Begriffsklärung

In diesem Abschnitt können Sie das Verhalten und die Vorlage für die eindeutige Identifizierung von Benutzereingaben konfigurieren, bevor Sie sie an die konfigurierte Wissensdatenbank senden.

- Disambiguierung aktivieren: Diese Option legt fest, ob Benutzereingaben vor dem Senden an die Wissensdatenbank eindeutig identifiziert werden sollen.

- Vorlage für Eingabeaufforderung zur Disambiguierung: Dies ist die Vorlage für die Eingabeaufforderung, die verwendet wird, um Benutzereingaben eindeutig zu kennzeichnen, wenn eine Verbindung zu einer Wissensdatenbank besteht. Die anhand dieser Eingabeaufforderung generierte Ausgabe wird als Abfrage verwendet, die an die Wissensdatenbank gesendet wird. Die Deaktivierung der Disambiguierung würde dazu führen, dass die Rohabfrage des Benutzers unverändert an die Wissensdatenbank gesendet wird.

Wenn die Disambiguierung aktiviert ist, würde beispielsweise eine nachfolgende Benutzerabfrage „Wie viel kostet das?“ könnte eindeutig mit „Wie viel kostet es, mein Nummernschild zu erneuern“ abgegrenzt werden? , was zu einer besseren Suchanfrage führt.

Verwenden Sie den bereitgestellten Text-Anwendungsfall

Die integrierte Benutzeroberfläche für den Text-Anwendungsfall soll es Geschäftsanwendern ermöglichen, die vom Admin-Benutzer erstellte Bereitstellung schnell zu erkunden und damit zu experimentieren. Vom Geschäftsbenutzer vorgenommene Konfigurationsänderungen werden nur für seine Sitzung wirksam. Der Geschäftsbenutzer muss diese Änderungen mit dem Administratorbenutzer teilen, der die Basisbereitstellung mit diesen Änderungen aktualisieren kann, damit alle sie verwenden können.

Die Chat-Benutzeroberfläche besteht aus den folgenden Komponenten:

- Chat-Fenster
- Chat-Eingabefeld
- Einstellungen
- Konversation löschen

Chat-Fenster

Speichert verschiedene Wendungen der Konversation. Nachrichten, die rechts beginnen, stammen vom Geschäftsbenutzer, und Nachrichten, die links beginnen, stammen vom konfigurierten LLM. Auf allen LLM-Antworten befindet sich ein kleines Zwischenablage-Symbol, um das einfache Kopieren von Antworten zu ermöglichen.

Chat-Eingabefeld

Am unteren Rand des Chat-Fensters befindet sich das Chat-Eingabefeld. Hier können Geschäftsanwender ihre Nachrichten eingeben, die an das LLM gesendet werden sollen. Direkt über dem Eingabefeld befindet sich der Verbindungsstatus. Wenn die Verbindung unterbrochen wird (z. B. aufgrund von Inaktivität), wird beim nächsten Senden einer Chat-Nachricht automatisch eine neue Verbindung hergestellt. Diese Anfrage wird aufgrund der zusätzlichen WebSocket Verbindungszeit voraussichtlich etwas länger dauern.

Je nach Konfiguration kann es sein, dass für die Eingabe eine maximale Länge erzwungen wird. Wenn dieses Limit überschritten wird, erhalten Benutzer eine Warnung und die Nachricht wird nicht gesendet.

Hinweis: Wenn Sie RAG mit Amazon Kendra verwenden, kürzt die Retrieve-API Abfragen auf 30 Token-Wörter. Wenn Sie längere Benutzereingaben erwarten, prüfen Sie, wie sich dies auf die Suchleistung auswirken könnte.

Einstellungen

Damit Geschäftsanwender schnell mit verschiedenen Konfigurationen experimentieren können, steht ein Einstellungsfenster zur Verfügung, in dem bestimmte Einrichtungskonfigurationsoptionen on-the-fly bearbeitet werden können

(Beispiel: Vorlage für Eingabeaufforderungen). Diese Änderungen können nur zu Beginn einer neuen Sitzung vorgenommen werden. Sobald eine Konversation gestartet wurde, ermöglicht das Löschen der Konversation die Bearbeitung der Konfigurationseinstellungen wieder.

Hinweis: Admin-Benutzer können wählen, ob die Einstellungen einer Bereitstellung gesperrt werden sollen. Sie können Live-Änderungen während der Bereitstellung mithilfe des Assistenten während der Eingabeaufforderung verhindern.

Klare Konversation

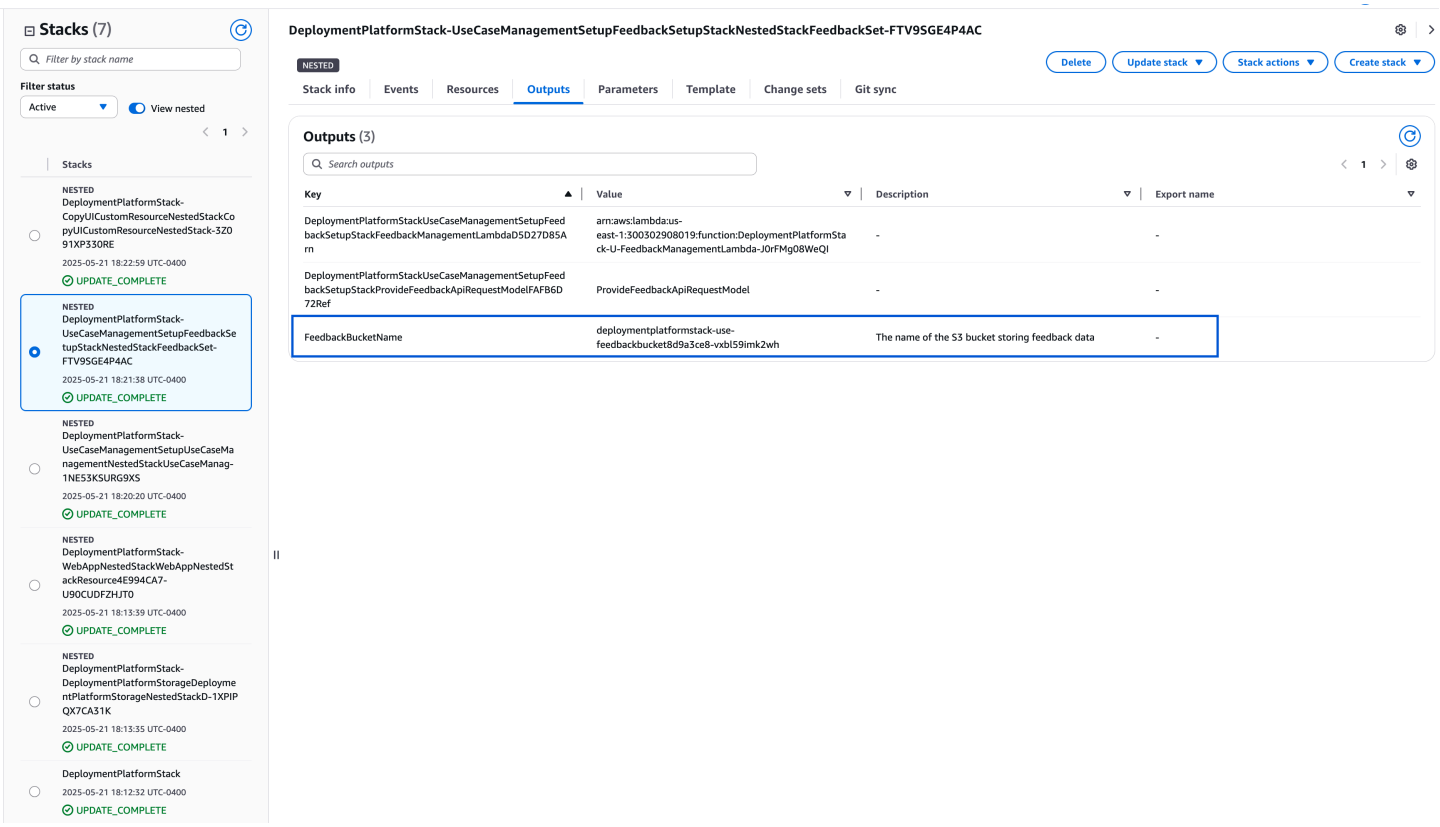
Im Verlauf der Konversation führt die Lösung einen Chat-Verlauf, der ein Konversationserlebnis ermöglicht. Dies ermöglicht die Disambiguierung von Abfragen und Folgefragen. Um eine Konversation zurückzusetzen und den gesamten Chatverlauf für diese Interaktion zu löschen, wählen Sie oben im Chatfenster *Konversation löschen*. Sobald die Konversation gelöscht wurde, wird eine neue Sitzung erstellt, die die Bearbeitung der Einstellungen wieder ermöglicht.

Zugriff auf und Analyse des vom Benutzer gesammelten Feedbacks

Ab Version 3.0.0 stellt das Deployment Dashboard einen verschachtelten Feedback-Stapel bereit, der es Text- und Bedrock Agent-Anwendungsfällen, die mit dem Dashboard bereitgestellt werden, ermöglicht, Feedback für die vom Dashboard generierten Antworten zu sammeln. LLM/Agent Insbesondere können Benutzer ein positives oder negatives Feedback zusammen mit einem optionalen Kommentar abgeben. Wenn der Benutzer ein negatives Feedback abgibt, kann er zusätzlich eine dieser negativen Kategorien auswählen: „Ungenau“, „Unvollständig oder unzureichend“, „Schädlich“, „Andere“. and/or

Sobald der Benutzer das Feedback gegeben hat, wird das Feedback in einem S3-Bucket gespeichert, der nach Anwendungsfall-ID, Jahr und Monat partitioniert ist. Die Anwendungsfall-ID befindet sich im Deployment Dashboard und der Feedback-S3-Bucket befindet sich in den Ausgaben des verschachtelten Feedback-Stacks des Deployment Dashboard-Stacks:

Zeigt den Bereitstellungsstapel an — der Name des Feedback-Buckets wird gefunden



Das Benutzerfeedback wird als API-Anfrage gesendet, die eine minimale Menge an Informationen enthält:

```
{
  "useCaseRecordKey": "a1b2c3d4-e5f6g7h8",
  "conversationId": "12345678-1234-1234-1234-123456789012",
  "messageId": "87654321-4321-4321-4321-210987654321",
  "rephrasedQuery": "What are the key features of the Generative AI Application Builder
 on AWS?",
  "sourceDocuments": [
    "s3://bucket-name/document1.pdf",
    "s3://bucket-name/document2.pdf"
  ],
  "feedback": "positive",
  "feedbackReason": [
    "Incomplete or insufficient"
  ],
  "comment": "The response was helpful but could include more details about important
 features."
}
```

Diese Nutzlast wird dann von einem Lambda verarbeitet, wobei der verwendet wird`useCaseRecordKey`, der die korrekte Konfiguration eines Anwendungsfalls zum Zeitpunkt der Bereitstellung identifiziert. Diese Konfiguration wird verwendet, um spezifische Details für das Feedback abzurufen, wie z. B. den ConversationTable Namen (enthält alle Konversationen sowie menschliche und KI-Nachrichtensequenzen), der dann verwendet wird, um das tatsächliche und abzurufen. `userInput` `llmResponse` Diesem Feedback-Datensatz sind auch zusätzliche Details beigefügt, z. B. das `agentId` und `agentAliasId` für einen Bedrock Agent-Anwendungsfall und usw. für einen Text-Anwendungsfall `modelProviderbedrockModelId`, der diese Konfiguration verwendet. Einzelheiten zum Zugriff auf diese Konfiguration finden Sie weiter unten im Abschnitt Benutzerdefinierte Feedback-Zuordnungen. Jede eingehende Feedback-Anfrage wird als JSON-Objekt gespeichert, und ein Beispiel für einen Feedback-Datensatz kann für einen Text-Anwendungsfall wie folgt aussehen:

```
{
   "useCaseId": "12345678-1234-1234-1234-123456789012",
   "useCaseRecordKey": "c07a2e3b-2f31b1e0",
   "userId": "22345678-1234-1234-1234-123456789012",
   "conversationId": "dd51de5d-5af1-4ec6-91d2-aadf14352109",
   "messageId": "32345678-1234-1234-1234-123456789012",
   "userInput": "What are its key features?",
   "rephrasedQuery": "What are the key features of the Generative AI Application
 Builder on AWS?",
```

```
   "llmResponse": "Generative AI Application Builder on AWS can help you build
 production ready enterprise chatbots rapidly.",
   "feedback": "negative",
   "feedbackReason": [
      "Incomplete or insufficient"
   ],
   "comment": "The response was helpful but could include more details about important
 features.",
   "timestamp": "2025-05-22T18:48:08.340Z",
   "feedbackId": "42345678-1234-1234-1234-123456789012",
   "useCaseType": "Text",
   "modelProvider": "Bedrock",
   "bedrockModelId": "amazon.nova-lite-v1:0",
   "ragEnabled": "false"
}
```

oder so für einen Bedrock Agent-Anwendungsfall:

```
{
   "useCaseId": "12345678-1234-1234-1234-123456789012",
   "useCaseRecordKey": "c07a2e3b-2f31b1e0",
   "userId": "22345678-1234-1234-1234-123456789012",
   "conversationId": "dd51de5d-5af1-4ec6-91d2-aadf14352109",
   "messageId": "32345678-1234-1234-1234-123456789012",
   "userInput": "What are its key features?",
   "llmResponse": "Generative AI Application Builder on AWS can help you build
 production ready enterprise chatbots rapidly.",
   "feedback": "negative",
   "feedbackReason": [
      "Incomplete or insufficient"
   ],
   "comment": "The response was helpful but could include more details about important
 features.",
   "timestamp": "2025-05-22T18:48:08.340Z",
   "feedbackId": "42345678-1234-1234-1234-123456789012",
   "useCaseType": "Agent",
   "agentId": "AHFXUJCAK1",
   "agentAliasId": "KSEDKOS0BL"
}
```

Dieses Feedback kann dann für die weitere Verarbeitung, Analyse und Modellierung von Wiederholungs- und Feedback-Schleifen verwendet werden. Sie können auch benutzerdefinierte

Zuordnungen hinzufügen, um den Feedback-Datensatz, der im Feedback-Lambda gespeichert wird, zu verbessern.

Benutzerdefinierte Feedback-Zuordnungen

Das Deployment Dashboard enthält eine`LLMConfigTable`, die in den Stack-Ausgaben des Deployment Dashboard-Stacks mit dem Schlüssel zu finden ist. `LLMConfigTableName` `LLMConfigTable`enthält die Konfigurationen für jeden Anwendungsfall auf der Grundlage der Einstellungen, die der Administrator bei der Bereitstellung des Anwendungsfalls über den Deployment-Dashboard-Assistenten ausgewählt hat. Jede Anwendungsfallkonfiguration wird anhand ihrer identifiziert. `useCaseRecordKey` Hier ist ein Beispiel für einen Konfigurationsdatensatz für einen Anwendungsfall in der: `LLMConfigTable`

```
{
   "key": "2dd76cfa-bc1a14da",
   "config": {
      "ConversationMemoryParams": {
        ...
      },
      "FeedbackParams": {
         "CustomMappings": {
            "NumberOfDocs": "$.KnowledgeBaseParams.NumberOfDocs",
            "ScoreThreshold": "$.KnowledgeBaseParams.ScoreThreshold"
         },
         "FeedbackEnabled": true
      },
      "IsInternalUser": "true",
      "KnowledgeBaseParams": {
         "KendraKnowledgeBaseParams": {
            "ExistingKendraIndexId": "d2831033-667f-4539-ab28-e6c7c7c5988b",
            "RoleBasedAccessControlEnabled": false
         },
         "KnowledgeBaseType": "Kendra",
         "NumberOfDocs": 5,
         "ReturnSourceDocs": false,
         "ScoreThreshold": 0.3
      },
      "LlmParams": {
         "BedrockLlmParams": {
            "BedrockInferenceType": "QUICK_START",
            "ModelId": "amazon.nova-lite-v1:0"
         },
```

```
            "ModelParams": {},
            "ModelProvider": "Bedrock",
            "PromptParams": {
               ...
            },
            "RAGEnabled": true,
            "Streaming": false,
            "Temperature": 0.1,
            "Verbose": false
         },
         "UseCaseName": "test-rag-usecase",
         "UseCaseType": "Text"
      }
}
```

Wenn Feedback für einen Anwendungsfall aktiviert ist, enthält diese Konfiguration ein Objekt, das es ermöglicht, dass ein `FeedbackParams` `CustomMappings` Objekt darin enthalten ist, das JSONPaths für alle zusätzlichen Felder angeben kann, dem Feedback-JSON-Datensatz hinzugefügt werden, der im Feedback-S3-Bucket gespeichert ist. Zum Beispiel für die obige Beispiel-Anwendungsfallkonfiguration CustomMappings enthält das Objekt `NumberOfDocs` und `ScoreThreshold` JSONPaths zusätzlich das `CustomMappings` Objekt, das mit dem `config` Stamm von beginnt. JSONPath Mit dieser Konfiguration erhält jeder JSON-Datensatz, der im Feedback-S3-Bucket gespeichert ist, diese beiden zusätzlichen Werte, abgesehen von den Feldern, die bereits bereitgestellt wurden.

Analysieren von Feedback-Daten

Die Feedback-Daten werden in S3 als JSON-Objekte gespeichert. Hier sind einige Ansätze, um diese Feedback-Daten zugänglicher und umsetzbarer zu machen:

Verwendung von AWS Glue und Amazon Athena

AWS Glue und Amazon Athena bieten eine serverlose Möglichkeit, Ihre Feedback-Daten zu katalogisieren, abzufragen und zu analysieren.

Mit AWS Glue können Sie einen AWS Glue Glue-Crawler erstellen, der die Daten in einem S3-Bucket untersucht, sein Schema ableitet und alle relevanten Metadaten in einem Katalog aufzeichnet. Danach können Dienste wie Amazon Athena verwendet werden, um die Daten abzufragen.

In der AWS Athena-Dokumentation finden Sie die Schritte zur Verbindung des Feedback-S3-Buckets mit Amazon Athena mithilfe des AWS Glue Glue-Datenkatalogs. Sie können auch einige der

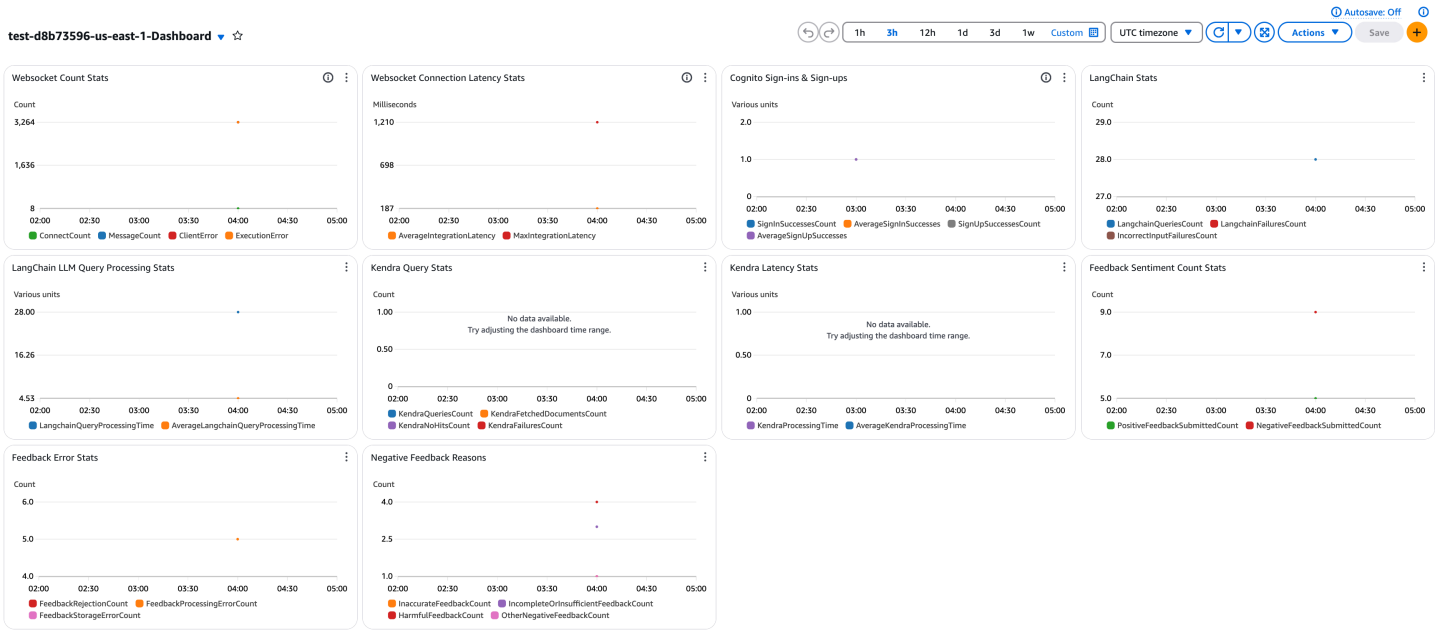
leistungsfähigeren Funktionen von Glue verwenden, um ETL-Jobs (Extract Transform & Load) mit diesen Daten durchzuführen und sie in ein Format umzuwandeln, das Ihren Anwendungsfällen für Analysen oder Modellumschulungen entspricht. Mit Glue können Sie beispielsweise die Datensätze mit bestimmten Feedbacktypen filtern, fehlende Informationen ausfüllen und diese Daten auch in einen anderen Speicherort laden, z. B. in einen anderen S3-Bucket oder einen anderen AWS-Datenspeicher.

> **Note**
>
> Abhängig von Ihrem Anwendungsfall sollten Sie erwägen, den Glue-Crawler so zu planen, dass er regelmäßig (z. B. wöchentlich) und nicht jede Nacht läuft, um die Kosten zu optimieren, da Feedback-Daten spärlich sein können.

Verwenden Sie die Dashboards der Lösung CloudWatch

Sie haben auch Zugriff auf ein im Lieferumfang der Lösung enthaltenes CloudWatch Dashboard, das Ihnen für jeden Anwendungsfall Trends für positives und negatives Feedback, Kategorien von Gründen für negatives Feedback usw. bietet. Sie finden dieses Dashboard anhand Ihres Anwendungsfallnamens unter Dashboards in der CloudWatch AWS-Konsole:

Stellt das Usecase-Dashboard dar CloudWatch



Sie können in diesem Dashboard auch zusätzliche Widgets erstellen oder Amazon Quick Sight-Dashboards erstellen.

Bewährte Methoden für die Analyse von Feedback-Daten

- Implementieren Sie Richtlinien für den Datenlebenszyklus in Ihrem S3-Bucket, um ältere Feedback-Daten auf kostengünstigeren Speicherebenen zu archivieren
- Erstellen Sie für jeden Anwendungsfall eine separate Analyse, um modellspezifische Verbesserungsmöglichkeiten zu identifizieren
- Legen Sie Feedback-Schwellenwerte fest, die Warnmeldungen auslösen, wenn negatives Feedback akzeptable Werte überschreitet
- Exportieren Sie regelmäßig wichtige Erkenntnisse, um sie mit Stakeholdern und Teams zur Modellverbesserung zu teilen

Betriebsmetriken für eine Bereitstellung anzeigen

Das Bereitstellungs-Dashboard und die Anwendungsfall-Stacks verfügen jeweils über ein eigenes CloudWatch Dashboard, in dem verschiedene Betriebskennzahlen der Lösung aufgezeichnet werden. Sie können diese CloudWatch Dashboards verwenden, um verschiedene Bereitstellungen zu vergleichen. So greifen Sie auf die Dashboards zu:

1. Navigieren Sie zur CloudWatch -Konsole.
2. Suchen Sie nach den vorgefertigten Dashboards, indem Sie entweder den Stacknamen oder den Universally Unique Identifier (UUID) nachschlagen.

Der Anwendungsfall Text enthält beispielsweise Diagramme, die die Anzahl der WebSocket Verbindungen, die Anzahl der Benutzeranmeldungen und -anmeldungen, die Zeit, die das LLM für die Bearbeitung eines Abschlusses benötigt hat, usw. aufzeichnen. Kunden können diese Grafiken verwenden, um verschiedene _quantitative_Kennzahlen einer Bereitstellung zu vergleichen.

Example

Es ist schwierig, die qualitativen Ergebnisse verschiedener Modelle zu vergleichen, die auf unterschiedliche Anwendungsfälle angewendet wurden. Verwenden Sie die Clone-Funktion, um schnell mehrere Bereitstellungen einzurichten, sodass Sie die Ergebnisse nebeneinander vergleichen können.

Zugriff auf CloudWatch Protokolle und Einblicke

Diese Lösung protokolliert Fehler-, Warn-, Informations- und Debuggingmeldungen für die Lambda-Funktionen. Um den Typ der zu protokollierenden Nachrichten auszuwählen:

1. Suchen Sie die entsprechende Funktion in der AWS Lambda Lambda-Konsole.
2. Fügen Sie eine Umgebungsvariable POWERTOOLS_LOG_LEVEL hinzu.
3. Stellen Sie die Variable auf den entsprechenden Nachrichtentyp ein.

Weitere Anweisungen finden Sie unter Create Lambda environment variables im AWS Lambda Developer Guide.

In der folgenden Tabelle sind die Typen von Protokollebenen aufgeführt, aus denen Sie wählen können.

Level	Description
FEHLER	Protokolle enthalten Informationen über alles, was dazu führt, dass ein Vorgang fehlschlägt.
WARNUNG	Protokolle enthalten Informationen über alles, was möglicherweise zu Inkonsistenzen in der Funktion führen könnte, aber nicht unbedingt zum Fehlschlagen des Vorgangs führen muss. Protokolle enthalten auch FEHLERMEL DUNGEN.
INFORMATIONEN	Die Protokolle enthalten allgemeine Informati onen darüber, wie die Funktion funktioniert. Die Protokolle enthalten auch FEHLER- und WARNMELDUNGEN.
DEBUGGEN	Protokolle enthalten Informationen, die beim Debuggen eines Problems mit der Funktion hilfreich sein können. Die Protokolle enthalten auch ERROR-, WARNING- und INFO-Meld ungen.

Gehen Sie wie folgt vor, um dieser Lösung CloudWatch Logs Insights hinzuzufügen.

1. Identifizieren Sie die relevanten Protokollgruppen:
   a. Melden Sie sich bei der CloudFormation AWS-Konsole an.
   b. Wählen Sie Ihren Ziel-Stack.
   c. Wählen Sie die Registerkarte Ressourcen und suchen Sie nach Ihren Lambda-Zielfunktionen.
   d. Melden Sie sich bei der AWS Lambda Lambda-Konsole an und wählen Sie jede Ihrer Lambda-Zielfunktionen aus.
   e. Wählen Sie für jede Ihrer Lambda-Zielfunktionen die Registerkarte Überwachen und dann CloudWatch Protokolle anzeigen aus.
   f. Kopieren Sie die Namen der Protokollgruppen, aus denen Sie Erkenntnisse extrahieren möchten.
2. Navigieren Sie zur CloudWatch Amazon-Konsole.
3. Wählen Sie im Navigationsmenü unter Logs die Option Logs Insights aus.
4. Wählen Sie auf der Seite Logs Insights den Tab Logs aus.
5. Suchen Sie nach den Namen der Protokollgruppen aus Schritt 1.
6. Kopieren Sie eine der folgenden Beispielabfragen und fügen Sie sie in das Abfragefeld ein:
   a. Um alle Client-Ausnahmen zu identifizieren:

```
fields @message
|filter @message like /(?i)Exception/|stats count(*) as exceptionCount by @message
```

   b. Um die Anzahl der Aufrufe nach Funktionsnamen abzurufen:

```
stats count(*) by function_name
```

   c. Um die Anzahl der Aufrufe in Intervallen von fünf Minuten abzurufen:

```
stats count(*) as invocations by bin(5m)
```

   d. So rufen Sie den gesamten AWS X-Ray-Trace ab IDs:

```
filter @message like "XRAY TraceId"
|parse @message "XRAY TraceId: * " as traceId|stats count(*) by traceId
```

   e. So rufen Sie Protokolle ab, die sich auf eine bestimmte X-Ray Trace ID beziehen:

```
filter @message like "your-traceid-here"
```

f. So rufen Sie nicht autorisierte WebSocket Fehler ab:

```
fields
@ingestionTime,
@log,
@logStream,
@message,
@requestId,
@timestamp,
errorMessage,
errorType
|filter @message like /Unauthorized/ and @message like /websocket/|sort @timestamp
 desc
```

g. So rufen Sie die Anzahl der veröffentlichten Metriken ab:

```
filter @message like "CloudWatchMetrics"
|parse @message /"Metrics":\s*\[(?<metrics>.*?)\]/|stats count(*) as metric_count
 by metrics
```

Entwicklerhandbuch

Dieser Abschnitt enthält den Quellcode für die Lösung, einen Integrationsleitfaden, einen Leitfaden zur Anpassung und eine API-Referenz.

Quellcode

Besuchen Sie unser GitHub Repository, um die Quelldateien für diese Lösung herunterzuladen und Ihre Anpassungen mit anderen zu teilen.

Die Generative AI Application Builder on AWS-Vorlagen werden mit dem AWS Cloud Development Kit (AWS CDK) generiert. Weitere Informationen finden Sie in der Datei README.md.

Leitfaden zur Integration

Die gesamte Lösung ist so konzipiert, dass sie leicht erweiterbar ist. Die Orchestrierungsebene dieser Lösung basiert auf. LangChain Sie können dieser Lösung jeden Modellanbieter, jede Wissensdatenbank oder jeden beliebigen Typ von Konversationsspeicher hinzufügen, der von LangChain (oder einem Drittanbieter, der LangChain Konnektoren für diese Komponenten bereitstellt) unterstützt wird.

Erweiterung wird unterstützt LLMs

Um einen weiteren Modellanbieter hinzuzufügen, z. B. einen benutzerdefinierten LLM-Anbieter, müssen Sie die folgenden drei Komponenten der Lösung aktualisieren:

1. Erstellen Sie einen neuen `TextUseCase` CDK-Stack, der die mit Ihrem benutzerdefinierten LLM-Anbieter konfigurierte Chat-Anwendung bereitstellt:
   a. Klonen Sie das GitHub Repository dieser Lösung und richten Sie Ihre Build-Umgebung ein, indem Sie den Anweisungen in der Datei README.md folgen.
   b. Kopieren Sie die `source/infrastructure/lib/bedrock-chat-stack.ts` Datei (oder erstellen Sie eine neue), fügen Sie sie in dasselbe Verzeichnis ein und benennen Sie sie um. `custom-chat-stack.ts`
   c. Benennen Sie die Klasse in der Datei in eine geeignete um, z. `CustomLLMChat` B.
   d. Sie können diesem Stack ein Secrets Manager Manager-Geheimnis hinzufügen, in dem Ihre Anmeldeinformationen für Ihr benutzerdefiniertes LLM gespeichert werden. Sie können diese

Anmeldeinformationen während des Modellaufrufs in der Chat-Lambda-Schicht abrufen, die im nächsten Absatz beschrieben wird.

2. Erstellen Sie eine Lambda-Schicht, die die Python-Bibliothek des hinzuzufügenden Modellanbieters enthält, und hängen Sie sie an. Für eine Chat-Anwendung für Amazon Bedrock-Anwendungsfälle enthält die `langchain-aws` Python-Bibliothek die benutzerdefinierten Konnektoren zusätzlich zum LangChain Paket, um eine Verbindung zu den AWS-Modellanbietern (Amazon Bedrock und SageMaker KI), Wissensdatenbanken (Amazon Kendra und Amazon Bedrock Knowledge Bases) und Speichertypen (wie DynamoDB) herzustellen. In ähnlicher Weise haben andere Modellanbieter ihre eigenen Konnektoren. Diese Ebene hilft Ihnen, die Python-Bibliothek dieses Modellanbieters anzuhängen, sodass Sie diese Konnektoren in der Chat-Lambda-Schicht verwenden können, die das LLM aufruft (Schritt 3). In dieser Lösung wird ein benutzerdefinierter Asset-Bundler verwendet, um Lambda-Schichten zu erstellen, die mithilfe von CDK-Aspekten angehängt werden. So erstellen Sie eine neue Ebene für die Bibliothek des Anbieters benutzerdefinierter Modelle:

   a. Navigieren Sie zu der `LambdaAspects` Klasse in der `source/infrastructure/lib/utils/lambda-aspects.ts` Datei.

   b. Folgen Sie den Anweisungen zur Erweiterung der Funktionalität der Lambda-Aspects-Klasse in der Datei (z. B. zum Hinzufügen der `getOrCreateLangchainLayer` Methode). Um diese neue Methode zu verwenden (zum Beispiel`getOrCreateCustomLLMLayer`), aktualisieren Sie auch die `LLM_LIBRARY_LAYER_TYPES` Aufzählung in der `source/infrastructure/lib/utils/constants.ts` Datei.

3. Erweitern Sie die `chat` Lambda-Funktion, um einen Builder, einen Client und einen Handler für den neuen Anbieter zu implementieren.

   Die `source/lambda/chat` enthält die LangChain Verbindungen für verschiedene LLMs sowie die unterstützenden Klassen, um diese LLMs zu erstellen. Diese unterstützenden Klassen folgen den Entwurfsmustern von Builder und Object Oriented, um das LLM zu erstellen.

   Jeder Handler (z. B.`bedrock_handler.py`) erstellt zuerst einen Client, überprüft die Umgebung auf erforderliche Umgebungsvariablen und ruft dann eine `get_model` Methode auf, um die LangChain LLM-Klasse abzurufen. Die Methode generate wird dann aufgerufen, um das LLM aufzurufen und seine Antwort abzurufen. LangChain unterstützt derzeit Streaming-Funktionen für Amazon Bedrock, aber nicht SageMaker KI. Je nach Streaming- oder Nicht-Streaming-Funktionalität wird der entsprechende WebSocket Handler (`WebsocketStreamingCallbackHandler`oder`WebsocketHandler`) aufgerufen,

um die Antwort mithilfe der Methode an die WebSocket Verbindung zurückzuschicken. `post_to_connection`

Der `clients/builder` Ordner enthält die Klassen, die beim Erstellen eines LLM Builders mithilfe des Builder-Musters helfen. Zunächst `use_case_config` wird a aus einem DynamoDB-Konfigurationsspeicher abgerufen, in dem die Details darüber gespeichert werden, welche Art von Wissensdatenbank, Konversationsspeicher und Modell erstellt werden sollen. Es enthält auch relevante Modelldetails wie Modellparameter und Eingabeaufforderungen. Der Builder hilft Ihnen dann dabei, die Schritte zum Erstellen einer Wissensdatenbank, zum Erstellen eines Konversationsspeichers zur Aufrechterhaltung des Konversationskontextes für LLM, zum Einstellen der entsprechenden LangChain Callbacks für Streaming- und Nicht-Streaming-Fälle und zum Erstellen eines LLM-Modells auf der Grundlage der bereitgestellten Modellkonfigurationen zu befolgen. Die DynamoDB-Konfiguration wird zum Zeitpunkt der Anwendungsfallerstellung gespeichert, wenn Sie einen Anwendungsfall über das Deployment-Dashboard bereitstellen (oder wenn er von den Benutzern in eigenständigen Anwendungsfall-Stack-Bereitstellungen ohne das Deployment-Dashboard bereitgestellt wird).

Der `clients/factories` Unterordner hilft bei der Festlegung des geeigneten Konversationsspeichers und der Wissensdatenbankklasse auf der Grundlage der LLM-Konfiguration. Dies ermöglicht eine einfache Erweiterung auf alle anderen Wissensdatenbanken oder Speichertypen, die Ihre Implementierung unterstützen soll.

Der `shared` Unterordner enthält spezifische Implementierungen von Knowledge Base und Conversation Memory, die vom Builder innerhalb der Factories instanziiert werden. Es enthält auch Amazon Kendra- und Amazon Bedrock Knowledge Base-Retriever, die innerhalb aufgerufen werden, LangChain um Dokumente für die RAG-Anwendungsfälle abzurufen, sowie Callbacks, die vom LLM-Modell verwendet werden. LangChain

Die LangChain Implementierungen verwenden LangChain Expression Language (LCEL), um Konversationsketten gemeinsam zu erstellen. `RunnableWithMessageHistory`class wird verwendet, um den Konversationsverlauf mit benutzerdefinierten LCEL-Ketten zu verwalten, sodass Funktionen wie das Zurücksenden von Quelldokumenten und die Verwendung der an die Wissensdatenbank gesendeten umformulierten (oder unmissverständlichen) Frage auch an das LLM gesendet werden können.

Um Ihre eigene Implementierung eines benutzerdefinierten Anbieters zu erstellen, können Sie:

a. Kopieren Sie die `bedrock_handler.py` Datei und erstellen Sie Ihren benutzerdefinierten Handler (z. B.`custom_handler.py`), der Ihren benutzerdefinierten Client erstellt (z. B.`CustomProviderClient`) (wie im folgenden Schritt angegeben).

b. Kopieren Sie `bedrock_client.py` in den Client-Ordner. Benennen Sie es um in `custom_provider_client.py` (oder Ihren spezifischen Modellanbieternamen, z. B.`CustomProvider`). Benennen Sie die darin enthaltene Klasse entsprechend, z. B. `CustomProviderClient` welche erbt`LLMChatClient`.

   Sie können die von bereitgestellten Methoden verwenden `LLMChatClient` oder Ihre eigenen Implementierungen schreiben, um diese zu überschreiben.

   Die `get_model` Methode erstellt eine `CustomProviderBuilder` (siehe den folgenden Schritt) und ruft die `construct_chat_model` Methode auf, die das Chat-Modell mithilfe von Builder-Schritten erstellt. Diese Methode fungiert im Builder-Muster als Director.

c. Kopieren Sie es `clients/builders/bedrock_builder.py` und benennen Sie es um `custom_provider_builder.py` und die darin enthaltene Klasse in die Klasse`CustomProviderBuilder`, die erbt LLMBuilder (`llm_builder.py`). Sie können die von bereitgestellten Methoden verwenden LLMBuilder oder Ihre eigenen Implementierungen schreiben, um diese zu überschreiben. Die Builder-Schritte werden nacheinander innerhalb der `construct_chat_model` Methode des Clients aufgerufen, z. B. `set_model_defaultsset_knowledge_base`, und`set_conversation_memory`.

   Die `set_llm_model` Methode würde das eigentliche LLM-Modell unter Verwendung aller Werte erstellen, die mit den zuvor aufgerufenen Methoden festgelegt wurden. Insbesondere können Sie ein RAG (`CustomProviderRetrievalLLM`) oder ein LLM ohne RAG (`CustomProviderLLM`) erstellen, das auf dem basiert, was aus der `rag_enabled variable` LLM-Konfiguration in DynamoDB abgerufen wurde.

   Diese Konfiguration wird in der Methode in der Klasse abgerufen. `retrieve_use_case_config` `LLMChatClient`

d. Implementieren Sie Ihre `CustomProviderLLM` oder `CustomProviderRetrievalLLM` - Implementierung im `llm_models` Unterordner, je nachdem, ob Sie einen RAG-Anwendungsfall oder einen anderen Anwendungsfall benötigen. Die meisten Funktionen zur Implementierung dieser Modelle werden in ihren jeweiligen `RetrievalLLM` Klassen für `BaseLangChainModel` Anwendungsfälle bereitgestellt, die nicht von RAG und RAG stammen.

Sie können die `llm_models/bedrock.py` Datei kopieren und die erforderlichen Änderungen vornehmen, um das LangChain Modell aufzurufen, das sich auf Ihren benutzerdefinierten Anbieter bezieht. Amazon Bedrock verwendet beispielsweise eine `ChatBedrock` Klasse, um ein Chat-Modell mithilfe von LangChain zu erstellen.

Die Generate-Methode generiert die LLM-Antwort mithilfe der LangChain LCEL-Ketten.

Sie können die `get_clean_model_params` Methode auch verwenden, um die Modellparameter gemäß LangChain Ihren Modellanforderungen zu bereinigen.

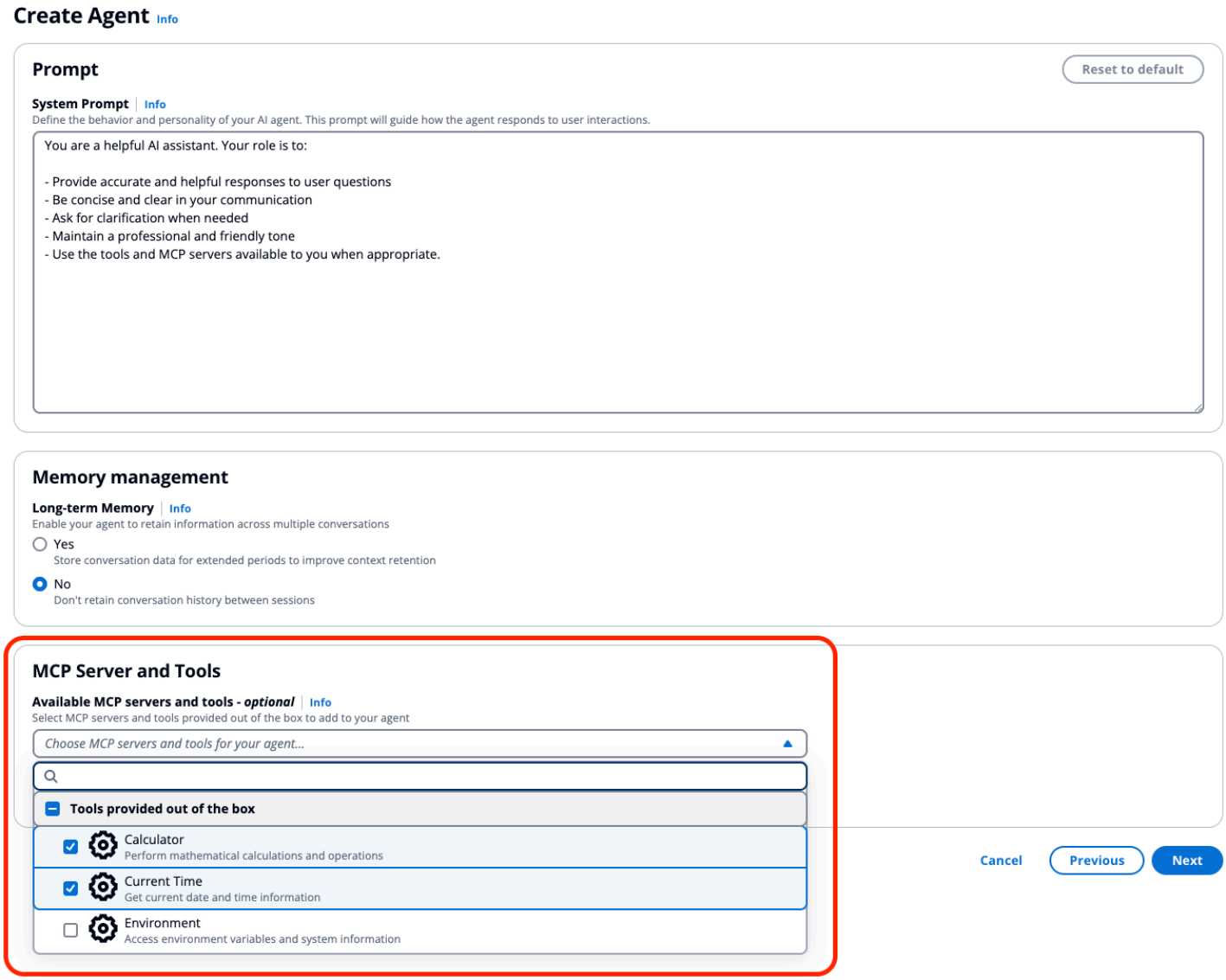
Erweiterung der unterstützten Tools von Strands

Mit der Lösung können Sie MCP-Server, KI-Agenten und Multi-Agent-Workflows erstellen und bereitstellen. Im Rahmen von Agent Builder können Sie MCP-Server anhängen, um Ihren Agenten zusätzliche Funktionen zu bieten. Zusätzlich zu den MCP-Servern können Sie die integrierten Tools von Strands (dem von der Lösung verwendeten zugrunde liegenden Framework) nutzen.

Die Lösung ist standardmäßig mit den folgenden Strons-Tools vorkonfiguriert:

- Aktuelle Uhrzeit (standardmäßig aktiviert)
- Taschenrechner (standardmäßig aktiviert)
- Umgebung

Auswahl des MCP-Servers und der Tools im Agent Builder-Assistenten mit integrierten Strands Tools



Um Ihre Agenten um zusätzliche Strons-Tools zu erweitern, folgen Sie dem in diesem Abschnitt beschriebenen vierstufigen Prozess.

Schritt 1: Finden Sie das Strans-Tool

Durchsuchen Sie die verfügbaren Strands Tools, um das Tool zu finden, das Sie verwenden möchten. Jedes Tool hat spezifische Funktionen und Konfigurationsanforderungen.

Um beispielsweise Funktionen zum Abrufen der Amazon Bedrock Knowledge Base hinzuzufügen, würden Sie das Abruftool verwenden.

Schritt 2: Aktualisieren Sie den SSM-Parameter

Um ein Tool in der Agent Builder-Bereitstellungsoberfläche verfügbar zu machen, aktualisieren Sie den AWS Systems Manager Parameter Store, der definiert, welche Strons-Tools unterstützt werden.

1. Navigieren Sie in Ihrem AWS-Konto zum AWS Systems Manager Parameter Store.
2. Suchen Sie den Parameter: `/gaab/<stack-name>/strands-tools`
3. Fügen Sie Ihre Werkzeugkonfiguration mithilfe der folgenden JSON-Struktur am Ende der vorhandenen Liste hinzu:

```
{
  "name": "Bedrock KB Retrieve",
  "description": "Retrieve information from Bedrock Knowledge Base",
  "value": "retrieve",
  "category": "AI",
  "isDefault": false
}
```

Feld	Description
name	Anzeigename, der in der Agent Builder-B enutzeroberfläche angezeigt wird
description	Kurze Beschreibung der Funktionen des Tools
value	Der genaue Werkzeugname, wie er im Strands Tool-Paket definiert ist
category	Organisatorische Kategorie für die Gruppieru ng von Tools in der Benutzeroberfläche
ist Standard	Ob das Tool standardmäßig für neue Agenten aktiviert werden soll

Schritt 3: Umgebungsvariablen konfigurieren

Viele Strands-Tools benötigen Umgebungsvariablen für die Konfiguration. Sie können diese Variablen auf zwei Arten setzen:

Option 1: Direkte Konfiguration zur AgentCore Laufzeit

Aktualisieren Sie den bereitgestellten Agenten direkt auf Amazon Bedrock AgentCore Runtime mit den erforderlichen Umgebungsvariablen.

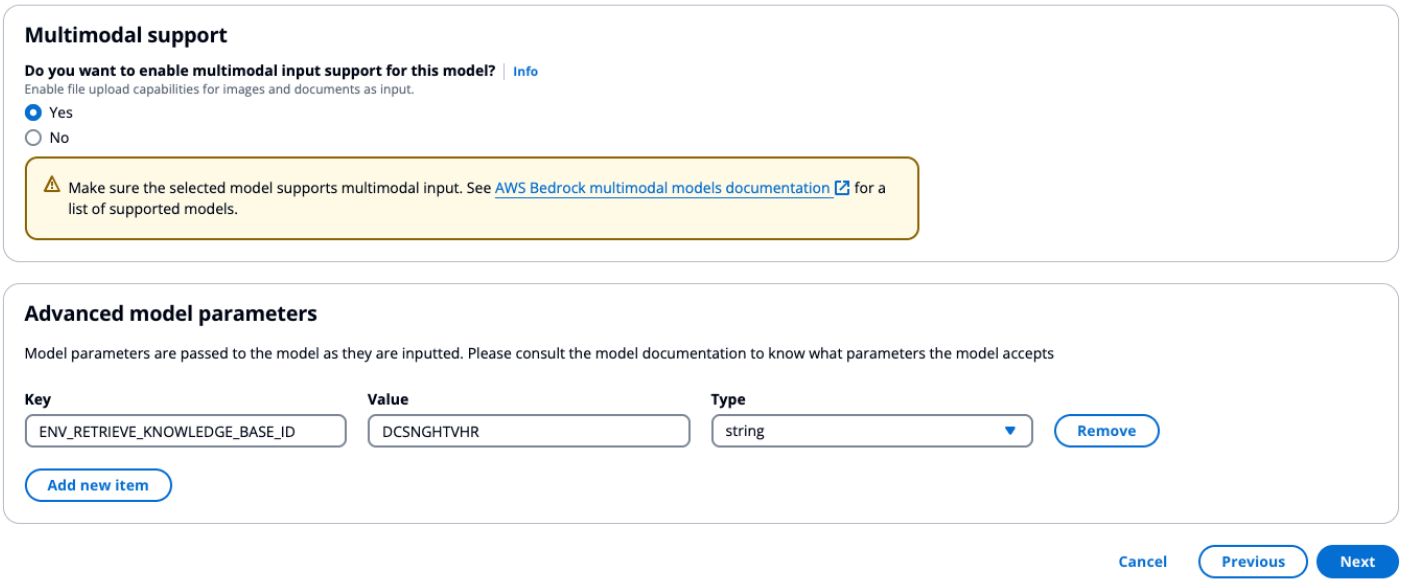
Option 2: Modellparameter im Bereitstellungsassistenten

Fügen Sie während des Schritts zur Modellauswahl im Agent Builder-Assistenten Umgebungsvariablen hinzu. Verwenden Sie dazu den Abschnitt Modellparameter. Umgebungsvariablen, die der Namenskonvention `ENV_<ALL_CAPS_TOOL_NAME>_<env_variable_name>` folgen, werden zur Laufzeit automatisch in die Ausführungsumgebung des Agenten geladen als`<env_variable_name>`.

Beispiel:

- `ENV_RETRIEVE_KNOWLEDGE_BASE_ID`wird `KNOWLEDGE_BASE_ID`
- `ENV_RETRIEVE_MIN_SCORE`wird `MIN_SCORE`

Abschnitt mit erweiterten Modellparametern, der die ENV_RETRIEVE_KNOWLEDGE_BASE_ID-Konfiguration zeigt



Informationen zu den erforderlichen Umgebungsvariablen finden Sie in der Dokumentation oder im Quellcode des jeweiligen Tools. Für das Tool zum Abrufen finden Sie die Konfigurationsoptionen im Quellcode.

Schritt 4: Fügen Sie IAM-Berechtigungen hinzu

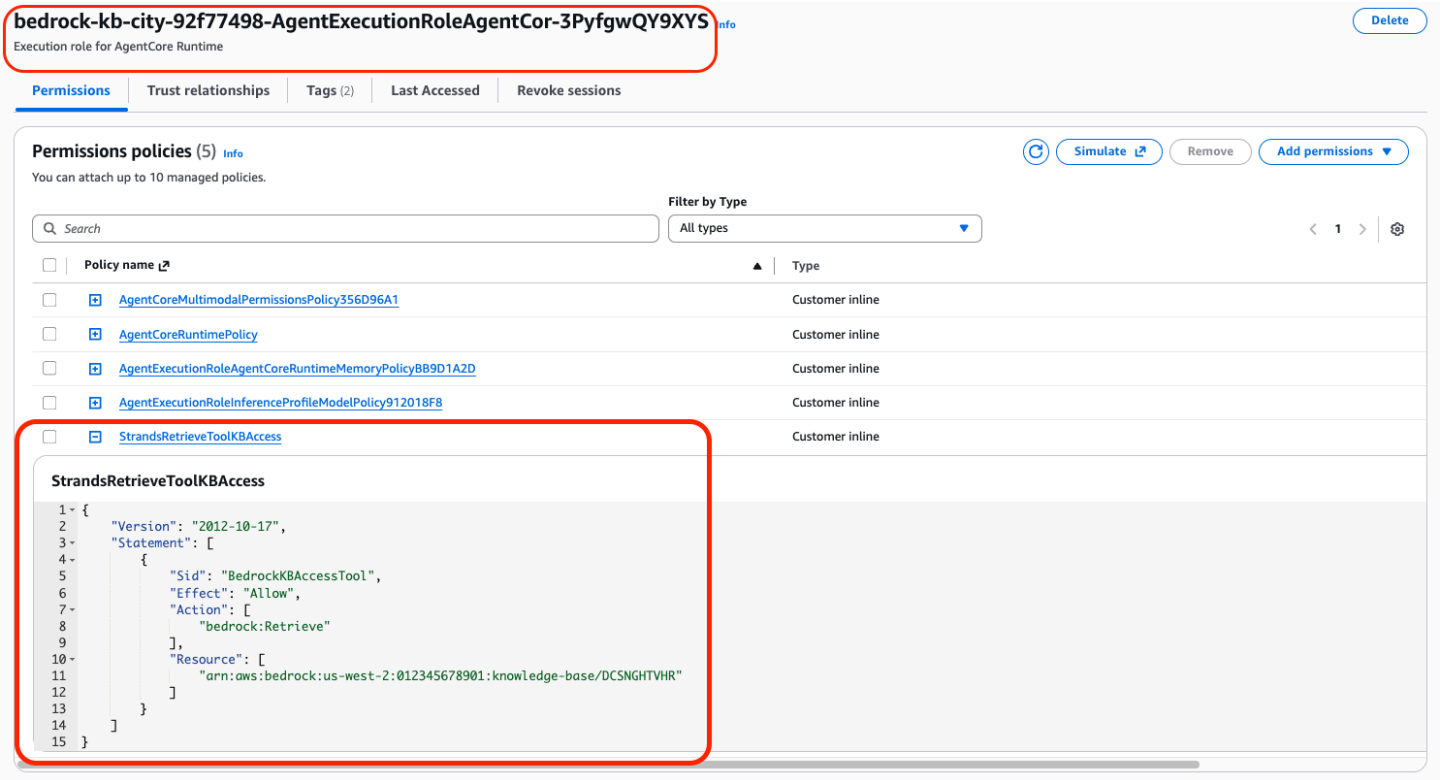
Fügen Sie Ihrer AgentCore Runtime-Ausführungsrolle manuell alle erforderlichen IAM-Berechtigungen hinzu, damit der Agent das Tool verwenden kann.

Um beispielsweise das Abruftool mit Amazon Bedrock Knowledge Bases zu verwenden:

1. Navigieren Sie in Ihrem AWS-Konto zur IAM-Konsole.
2. Suchen Sie die AgentCore Runtime-Ausführungsrolle für Ihren Agenten.
3. Fügen Sie die folgende Berechtigung hinzu:

```
{
  "Effect": "Allow",
  "Action": "bedrock:Retrieve",
  "Resource": "arn:aws:bedrock:region:account-id:knowledge-base/knowledge-base-id"
}
```

Die IAM-Konsole zeigt die StrandsRetrieveTool KBAccess Richtlinie an, die der AgentCore Runtime-Ausführungsrolle zugeordnet ist

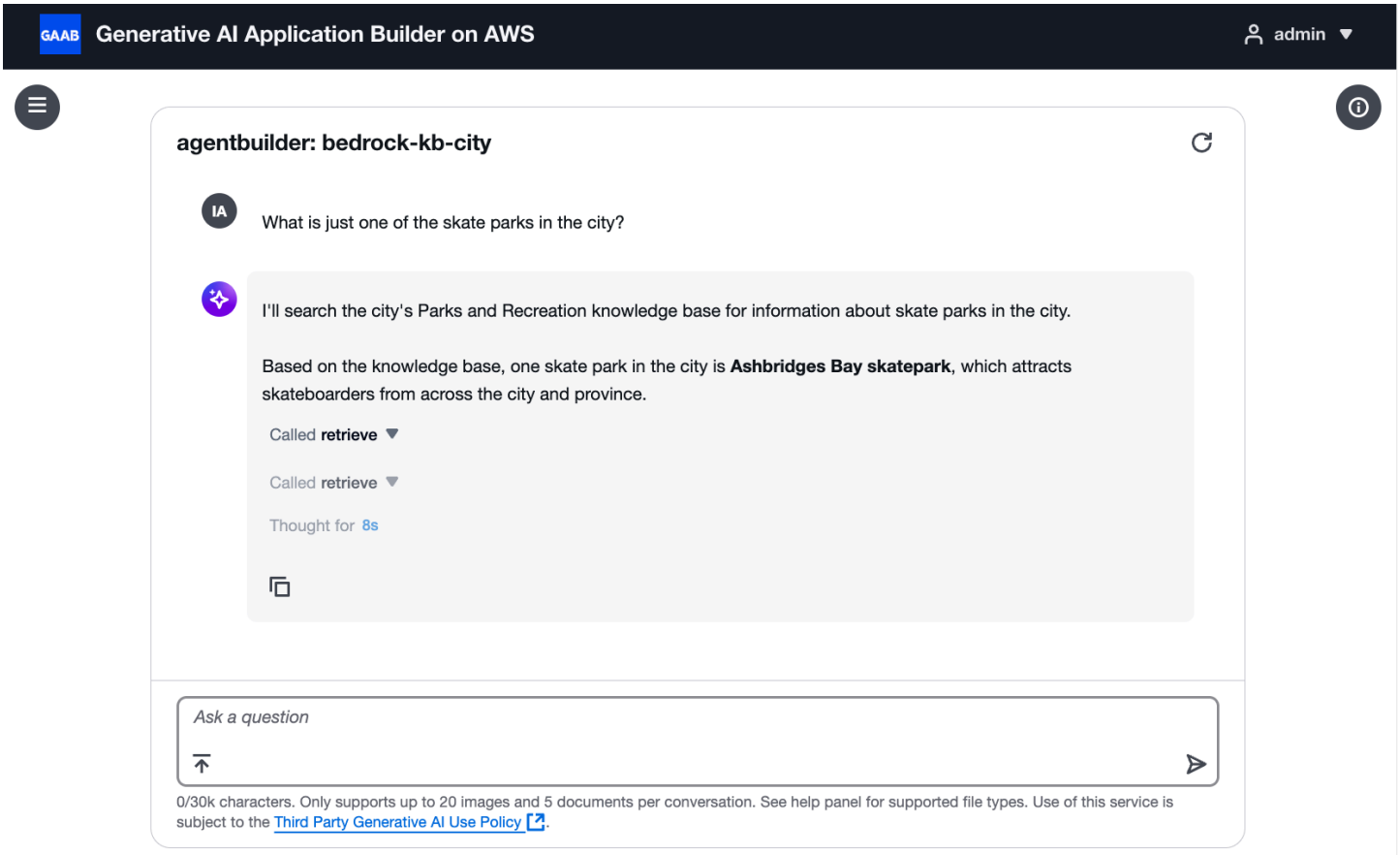


Welche spezifischen Berechtigungen erforderlich sind, hängt vom jeweiligen Tool ab. Schlagen Sie in der Dokumentation des Tools und in der AWS-Servicedokumentation nach, um die entsprechenden IAM-Berechtigungen zu ermitteln.

Schritt 5: Testen Sie den Agenten

Nachdem Sie die Konfigurationsschritte abgeschlossen haben, testen Sie Ihren Agenten, um sicherzustellen, dass das Tool ordnungsgemäß funktioniert. Sie sollten die Tool-Aufrufe in den Ausführungsprotokollen und Antworten des Agenten sehen.

Der Agent verwendet erfolgreich das Retrieve-Tool, um eine Frage zu Skateparks zu beantworten



> **Note**
>
> Eine vollständige Liste der verfügbaren Strands-Tools und ihrer Funktionen finden Sie in der Dokumentation zu den Strands Community Tools.

Erweiterung der unterstützten Wissensdatenbanken und Typen von Konversationsspeichern

Um Ihre Implementierungen von Conversation Memory oder Knowledge Base hinzuzufügen, fügen Sie die erforderlichen Implementierungen im `shared` Ordner hinzu und bearbeiten Sie dann die Factories und die entsprechenden Aufzählungen, um eine Instanz dieser Klassen zu erstellen.

Wenn Sie die LLM-Konfiguration angeben, die im Parameterspeicher gespeichert ist, werden der entsprechende Konversationsspeicher und die entsprechende Wissensdatenbank für Ihr LLM erstellt. Wenn zum Beispiel für DynamoDB angegeben `ConversationMemoryType` ist, wird eine Instanz von `DynamoDBChatMessageHistory` (available in`shared_components/memory/ddb_enhanced_message_history.py`) erstellt. Wenn Amazon Kendra angegeben `KnowledgeBaseType` ist, wird eine Instanz von `KendraKnowledgeBase` (innerhalb verfügbar`shared_components/knowledge/kendra_knowledge_base.py`) erstellt.

Erstellung und Bereitstellung der Codeänderungen

Erstellen Sie das Programm mit dem `npm run build` Befehl. Sobald alle Fehler behoben sind, führen Sie den Befehl aus, `cdk synth` um die Vorlagendateien und alle Lambda-Assets zu generieren.

1. Sie können das `0/stage-assets.sh` Skript verwenden, um alle generierten Assets manuell im Staging-Bucket in Ihrem Konto bereitzustellen.
2. Verwenden Sie den folgenden Befehl, um die Plattform bereitzustellen oder zu aktualisieren:

   ```
   cdk deploy DeploymentPlatformStack --parameters AdminUserEmail='admin-
   email@amazon.com'
   ```

   Alle zusätzlichen CloudFormation AWS-Parameter sollten ebenfalls zusammen mit dem AdminUserEmailParameter angegeben werden.

Leitfaden zur Anpassung

Verwaltung des Cognito-Benutzerpools

Wenn das Deployment-Dashboard bereitgestellt wird, werden ein Amazon Cognito Cognito-Benutzerpool zusammen mit einem Admin-Benutzer erstellt, um die Authentifizierung für die

Anwendung bereitzustellen. Dieser Benutzerpool wird im Deployment-Dashboard und in allen Anwendungsfällen gemeinsam genutzt. Dem Admin-Benutzer, der bei der Bereitstellung des Dashboards erstellt wurde, wird automatisch Zugriff auf alle Anwendungsfälle gewährt, die über das Dashboard bereitgestellt werden. Dieser Mechanismus wird über Amazon Cognito Cognito-Benutzerpoolgruppen bereitgestellt.

Wenn ein Anwendungsfall über das Dashboard bereitgestellt wird und eine E-Mail bereitgestellt wird, wird im gemeinsam genutzten Benutzerpool ein Benutzer erstellt, zusammen mit einer Benutzergruppe, die nach dem spezifischen Anwendungsfall benannt ist. Der neu erstellte Benutzer wird dann der Gruppe hinzugefügt, wodurch der Benutzer Zugriff auf den Anwendungsfall erhält.

Wenn Sie einem bestimmten Anwendungsfall einen zusätzlichen Benutzer hinzufügen möchten, können Sie dies erreichen, indem Sie einen Benutzer im Cognito-Benutzerpool erstellen und ihn zu den Gruppen hinzufügen, die den Anwendungsfällen entsprechen, auf die der Benutzer Zugriff haben soll. Eine step-by-step Anleitung finden Sie unter Einen neuen Benutzer in der AWS-Managementkonsole erstellen.

Ebenso müssen Sie, wenn Sie zusätzliche Admin-Benutzer erstellen möchten, einen neuen Benutzer erstellen und ihn der Admin-Gruppe im Benutzerpool hinzufügen.

Die Benutzernamen werden erstellt, indem der Teil der bereitgestellten E-Mail vor dem und die @ generierte UUID für den Anwendungsfall (oder `-admin` im Fall des Admin-Benutzers) angehängt wird.

Auf der Registerkarte Gruppen können Sie sehen, dass anhand des Namens des Anwendungsfalls (wie im Assistenten angegeben) und der UUID des Anwendungsfalls automatisch eine Admin-Gruppe und eine Gruppe für jeden Anwendungsfall erstellt wurden.

API-Referenz

Dieser Abschnitt enthält API-Referenzen für die Lösung.

Bereitstellungs-Dashboard

REST-API	HTTP-Methode	Funktionalität	Autorisierte Anrufer
`/deployments`	`GET`	Holen Sie sich alle Bereitstellungen.	Authentifiziertes JWT-Token von Amazon Cognito

REST-API	HTTP-Methode	Funktionalität	Autorisierte Anrufer
`/deployments`	`POST`	Erstellt eine neue Anwendungsfall-Bereitstellung.	Authentifiziertes JWT-Token von Amazon Cognito
`/deployments/{useCaseId}`	`GET`	Ruft Bereitstellungsdetails für eine einzelne Bereitstellung ab.	Authentifiziertes JWT-Token von Amazon Cognito
`/deployments/{useCaseId}`	`PATCH`	Aktualisiert eine bestimmte Bereitstellung.	Authentifiziertes JWT-Token von Amazon Cognito
`/deployments/{useCaseId}`	`DELETE`	Löscht eine bestimmte Bereitstellung.	Authentifiziertes JWT-Token von Amazon Cognito
`/model-info/use-case-types`	`GET`	Ruft die verfügbaren Anwendungsfalltypen für die Bereitstellung ab	Authentifiziertes JWT-Token von Amazon Cognito
`/model-info/{useCaseType}/providers`	`GET`	Ruft die verfügbaren Modellanbieter für den angegebenen Anwendungsfalltyp ab	Authentifiziertes JWT-Token von Amazon Cognito
`/model-info/{useCaseType}/{providerName}`	`GET`	Ruft die IDs Modelle ab, die für einen bestimmten Anbieter und einen bestimmten Anwendungsfalltyp verfügbar sind	Authentifiziertes JWT-Token von Amazon Cognito

REST-API	HTTP-Methode	Funktionalität	Autorisierte Anrufer
`/model-info/{useCaseType}/{providerName}/{modelId}`	`GET`	Ruft die Informationen über das angegebene Modell ab, einschließlich der Standardparameter.	Authentifiziertes JWT-Token von Amazon Cognito

> **Note**
>
> OpenAPI- und Swagger-Dateien können auch aus API Gateway exportiert werden, um die Integration mit der API zu vereinfachen. Siehe Exportieren einer REST-API aus dem API Gateway.

POST- und PATCH-Payloads

Im Folgenden finden Sie ein Beispiel für eine POST-Payload an den `/deployments` Endpunkt, wodurch ein neuer Anwendungsfall entsteht.

```
{
 "UseCaseName": "usecase1",
 "UseCaseDescription": "Description of the use case to be deployed. For display
 purposes", // optional
 "DefaultUserEmail": "placeholder@example.com", // optional, if not provided, the
 Cognito Group and User will not be created
 "DeployUI": true, // optional
 "VpcParams": {
 "VpcEnabled": true,
 "CreateNewVpc": false,
 // provide these if not creating new vpc
 "ExistingVpcId": "vpc-id",
 "ExistingPrivateSubnetIds": ["subnet-1", "subnet-2"],
 "ExistingSecurityGroupIds": ["sg-1", "sg-2"]
 },
 "ConversationMemoryParams": {
 "ConversationMemoryType": "DynamoDB",
 "HumanPrefix": "user", // optional
 "AiPrefix": "ai", // optional
 "ChatHistoryLength": 10 // optional
```

```
 },
 "KnowledgeBaseParams": {
 "KnowledgeBaseType": "Bedrock",
 // one of the following based on selected provider
 "BedrockKnowledgeBaseParams": {
 "BedrockKnowledgeBaseId": "my-bedrock-kb",
 "RetrievalFilter": {}, // optional
 "OverrideSearchType": "HYBRID" // optional
 },
 "KendraKnowledgeBaseParams": {
 "AttributeFilter": {}, // optional
 "RoleBasedAccessControlEnabled": true, // optional
 "ExistingKendraIndexId": "12345678-abcd-1234-abcd-1234567890ab",
 // provide the following in place of ExistingKendraIndexId if you want the solution to
 deploy an index for you
 "KendraIndexName": "index",
 "QueryCapacityUnits": 1, // optional
 "StorageCapacityUnits": 1, // optional
 "KendraIndexEdition": "DEVELOPER" // optional
 },
 "NoDocsFoundResponse": "Sorry, I couldn't find any relevant information for your
 query.", // optional
 "NumberOfDocs": 3, // optional
 "ScoreThreshold": 0.7, // optional
 "ReturnSourceDocs": true // optional
 },
 "LlmParams": {
 "ModelProvider": "Bedrock | SAGEMAKER",
 // one of the following based on selected provider
 "BedrockLlmParams": {
 "ModelId": "model-id", // use this for on demand models. Can't use with ModelArn
 "ModelArn": "model-arn", // use this for provisioned/custom models. Can't use with
 ModelId,
 "InferenceProfileId": "profile-id"
 "GuardrailIdentifier": "arn:aws:bedrock:us-east-1:123456789012:guardrail/my-
guardrail", // optional
 "GuardrailVersion": "1" // optional. Required if GuardrailIdentifier provided.
 },
 "SageMakerLlmParams": {
 "EndpointName": "some-endpoint",
 "ModelInputPayloadSchema": {},
 "ModelOutputJSONPath": "$."
 },
 // optional. Passes on arbitrary params to the underlying LLM.
```

```
 "ModelParams": {
 "param1": {
 "Value": "value1",
 "Type": "string"
 },
 "param2": {
 "Value": 1,
 "Type": "integer"
 }
 },
 // optional
 "PromptParams": {
 "PromptTemplate": "some template",
 "UserPromptEditingEnabled": true,
 "MaxPromptTemplateLength": 1000,
 "MaxInputTextLength": 1000,
 "DisambiguationPromptTemplate": "some disambiguation template",
 "DisambiguationEnabled": true
 },
 "Temperature": 1.0, // optional
 "Streaming": true, // optional
 "RAGEnabled": true, // optional. Must be true if providing KnowledgeBaseParams above.
 "Verbose": false // optional
 },
 "AgentParams": {
 "AgentType": "Bedrock",
 "BedrockAgentParams": {
 "AgentId": "agent-id",
 "AgentAliasId": "alias-id",
 "EnableTrace": true
 }
 },
 // optional
 "AuthenticationParams": {
 "AuthenticationProvider": "Cognito",
 "CognitoParams": {
 "ExistingUserPoolId": "user-pool-id",
 "ExistingUserPoolClientId": "client-id" // optional. If not provided, the solution
 will create a client for you in the provided pool
 }
 }

}
```

Für Updates ist die Struktur dieselbe wie oben, mit einigen Einschränkungen:

- Der Name des Anwendungsfalls kann nicht geändert werden
- Ein Anwendungsfall kann Sicherheitsgruppen und Subnetze erst ändern, wenn er in einer VPC bereitgestellt wurde. Die VPC selbst kann nicht geändert werden.
- Wenn ein Kendra-Index für Sie als Wissensdatenbank erstellt wurde, können Sie die Konfiguration dieses Indexes nicht ändern (z. B. KendraIndexName, QueryCapacityUnits)

Gemeinsamer Anwendungsfall APIs

Die folgenden REST-API-Endpunkte sind sowohl für Text- als auch für Bedrock Agent-Anwendungsfälle verfügbar:

REST-API	HTTP-Methode	Funktionalität	Autorisierte Anrufer
`/details/{useCaseConfigKey}`	`GET`	Ruft Konfigurationsdetails für einen bestimmten Anwendungsfall ab.	Authentifiziertes JWT-Token von Amazon Cognito

WebSocket API	Funktionalität	Autorisierte Anrufer
`/$connect`	WebSocket Verbindung herstellen und Benutzer authentifizieren.	Authentifiziertes JWT-Token von Amazon Cognito
`/$disconnect`	Endpunkt, der aufgerufen wird, wenn eine WebSocket Verbindung getrennt wurde.	Authentifiziertes JWT-Token von Amazon Cognito

Verwenden Sie die API mit Falldetails

Der Details-API-Endpunkt ruft Informationen zu einem bestimmten Anwendungsfall ab:

```
GET /details/{useCaseConfigKey}
```

Dieser Endpunkt gibt die Konfigurationsdetails für einen bestimmten Anwendungsfall zurück, einschließlich Modellparameter, Knowledgebase-Einstellungen und anderer Bereitstellungsinformationen. Für die Autorisierung ist ein von Amazon Cognito authentifiziertes JWT-Token erforderlich.

Anwendungsfall im Textformat

WebSocket API	Funktionalität	Autorisierte Anrufer
`/sendMessage`	Sendet die Chat-Nachricht des Benutzers WebSocket zur Verarbeitung mit der konfiguri erten LLM-Erfahrung an den.	Authentifiziertes JWT-Token von Amazon Cognito

REST-API	HTTP-Methode	Funktionalität	Autorisierte Anrufer
`/feedback/{useCaseId}`	`POST`	Sendet Benutzerf eedback für einen bestimmten Anwendungsfall.	Authentifiziertes JWT-Token von Amazon Cognito

Nutzlasten von SendMessage

Wenn Sie die `/sendMessage` API direkt integrieren, müssen Sie die folgenden Payload-Formate für Anfrage und Antwort einhalten.

Payload anfordern

```
{
 "action": "sendMessage",
 "question": "the message to send to the api",
 "conversationId": "", // If not provided, a new conversation will be created, with the
 conversationId returned in the response. All subsequent messages in that conversation
 (where history is retained), should provide the conversationId there.
 "promptTemplate": "", // Optional. Overrides the configured prompt
 "authToken": "XXXX" // Optional. accessToken from cognito flow. Required for RAG with
 RBAC
```

```
}
```

Name des Parameters	Typ	Description
Aktion	`String`	Derzeit unterstützen wir nur die Aktion „SendMessage“ auf der WebSocket
Frage	`String`	Die Benutzereingabe, die an das LLM gesendet werden soll
Konversations-ID	`String`	Eine UUID, die die Konversation identifiziert. Falls nicht angegeben, wird eine neue Konversation erstellt, wobei die ConversationID in der Antwort zurückgegeben wird. Alle nachfolgenden Nachrichten in dieser Konversation (in denen Sie möchten, dass history/context sie gespeichert werden), sollten dort die ConversationID angeben.
Vorlage für Eingabeaufforderung	`String` [Optional]	Überschreibt die Eingabeaufforderungsvorlage für diese Nachricht. Falls leer oder nicht angegeben, wird standardmäßig die bei der Bereitstellung festgelegte Aufforderung verwendet. Es müssen die richtigen Platzhalter für die angegebene Konfiguration angegeben werden (d. h. {history} und {input} für Sagemaker AI-Bereitstellungen, die nicht von RAG

Name des Parameters	Typ	Description
		stammen, mit dem Zusatz {context}, wenn RAG für alle Bereitstellungen verwendet wird.
AuthToken	`String` [Optional]	AccessToken, wie es aus dem Cognito-Authentifizierungsf luss abgerufen wurde. Dies ist erforderlich, wenn ein Chat-Websocket-Endpunkt aufgerufen wird, der für RAG mit Role Based Access Control (RBAC) konfiguriert ist. Die Cognito:groups-Ans pruchsliste in diesem JWT-Token wird verwendet, um den Zugriff auf Dokumente im Kendra-Index zu kontrolli eren. Dieser Parameter ist für Anwendungsfälle, die nicht RAG sind, nicht erforderlich. Er ist auch nicht für RAG-Anwendungsfälle erforderlich, bei denen RBAC deaktiviert ist.

Payloads für Antworten

Frage & Antwort

Die WebSocket API antwortet mit einem (wenn Streaming deaktiviert ist) oder vielen (wenn Streaming aktiviert ist) JSON-Objekten, die für jede Abfrage wie folgt strukturiert sind.

```
{
 "data": "some data",
 "conversationId": "id",
```

```
}
```

Name des Parameters	Typ	Description
data	`String`	Ein Teil der Antwort des LLM, wenn Streaming aktiviert ist, oder die gesamte Antwort. Bei Verwendung von Streaming wird eine Antwort dieses Formats mit dem Dateninha lt END_CONVERSATION gesendet, um das Ende der Antwort auf eine einzelne Frage anzuzeigen.
Konversations-ID	`String`	Die ID der Konversation, zu der diese SourceDocument-Antwort gehört.

Antwort auf das Quelldokument

Wenn Sie Ihren RAG-Anwendungsfall für die Rückgabe von Quelldokumenten konfiguriert haben, erhalten Sie am Ende jeder Antwort für jedes Quelldokument, das zur Erstellung der Antwort verwendet wurde, außerdem die folgende Payload.

```
{
 "sourceDocument": {
 "excerpt": "some excerpt from the",
 "location": "s3://fake-bucket/test.txt",
 "score": 0.500,
 "document_title": null,
 "document_id": null,
 "additional_attributes": null
 },
 "conversationId": "some-id"
}
```

Name des Parameters	Typ	Description
Auszug	`String`	Ein Auszug aus dem Quelldokument.
location	`String`	Speicherort des Quelldokuments. Dies hängt von den verwendeten Datenquellen und der Art der Wissensdatenbank ab, kann aber auch Dinge wie S3 URIs oder Websites sein.
Ergebnis	`Number` | `String`	Die Gewissheit, dass das Dokument der gestellten Frage entspricht. Dies wird ein Float von 0 bis 1 für Bedrock und eine Zeichenfolge (z. B. HIGH, LOW usw.) für Kendra sein.
document_title	`String`	Titel des zurückgegebenen Quelldokuments. Nur verfügbar, wenn Sie Kendra verwenden.
document_id	`String`	ID des zurückgegebenen Quelldokuments. Nur verfügbar, wenn Sie Kendra verwenden.
additional_attributes	`String`	Dieses Feld enthält alle zusätzlichen Attribute des Dokuments, wie sie bei der Aufnahme in Ihrer Wissensdatenbank angepasst wurden.

Name des Parameters	Typ	Description
Konversations-ID	`String`	Die ID der Konversation, zu der diese SourceDocument-Antwort gehört.

Nutzlast der Feedback-API

Im Folgenden finden Sie ein Beispiel für eine POST-Nutzlast an den `/feedback/{useCaseId}` Endpunkt, über die Benutzerfeedback für einen bestimmten Anwendungsfall gesendet wird:

```
{
  "useCaseRecordKey": "12345678-12345678",
  "conversationId": "12345678-1234-1234-1234-123456789012",
  "messageId": "12345678-1234-1234-1234-123456789012",
  "feedback": "positive",
  "feedbackReason": ["accurate", "helpful"],
  "comment": "This response was very helpful.",
  "rephrasedQuery": "What are the key features of Amazon Bedrock?",
  "sourceDocuments": [
    "s3://bucket-name/document1.pdf",
    "s3://bucket-name/document2.pdf"
  ]
}
```

Anwendungsfall Bedrock Agent

WebSocket API	Funktionalität	Autorisierte Anrufer
`/invokeAgent`	Sendet die Nachricht des Benutzers an den WebSocket zur Verarbeitung mit dem konfigurierten Agenten.	Authentifiziertes JWT-Token von Amazon Cognito

Agent-Nutzlasten aufrufen

Wenn Sie direkt mit dem integrieren`/invokeAgent API`, müssen Sie die folgenden Payload-Formate für Anfrage und Antwort einhalten.

Anforderungs-Nutzlast

```
{
"action": "invokeAgent",
"inputText": "User query to the agent",
"conversationId": "", // Optional. Empty conversationId implies a new conversation.
 When not provided, a new conversationId will be created and returned with the
 response. All subsequent messages in the same conversation should provide the same
 conversationId (i.e. chat memory/history is maintained).
"authToken": "XXXX" // Optional. accessToken from cognito flow. If provided, it needs
 to be a valid JWT token associated with the user
}
```

Parametername	Typ	Description
Aktion	`String`	Wir unterstützen nur die `invokeAgent` Aktion am WebSocket.
Text eingeben	`String`	Die Benutzereingabe, die an das LLM gesendet werden soll.
Konversations-ID	`String[Optional]`	Eine UUID, die die Konversation eindeutig identifiziert. Wenn Sie diesen Wert nicht angeben, erstellt die Lösung eine neue Konversation und gibt die ConversationID in der Antwort zurück. Alle nachfolgenden Nachrichten in dieser Konversation (in der Sie Verlauf und Kontext beibehalten möchten) geben dort die ConversationID an.
AuthToken	`String[Optional]`	AccessToken, wie es aus dem Amazon Cognito Cognito-Authentifizierungsablauf

Parametername	Typ	Description
		abgerufen wurde. Dieser Parameter ist nicht erforderl ich. Wenn Sie es angeben, wird das JWT-Token validiert . Dies erleichtert die Erweiteru ng dieser Lösung.

Payloads für Antworten

Antwort auf die Frage

Die WebSocket API antwortet mit einem (wenn Streaming deaktiviert ist) oder vielen (wenn Streaming aktiviert ist) JSON-Objekten, die für jede Abfrage wie folgt strukturiert sind.

```
{
 "data" "some data",
 "conversationId": "id",
 }
```

Parametername	Typ	Description
data	`String`	Die Antwort vom Agentenau fruf.
Konversations-ID	`String`	Die ID der Konversation.

Referenz

Dieser Abschnitt enthält Informationen zur Datenerfassung für diese Lösung, Hinweise auf verwandte Ressourcen und eine Liste der Entwickler, die zu dieser Lösung beigetragen haben.

Unterstützte LLM-Anbieter

Die Lösung kann in die folgenden LLM-Anbieter integriert werden:

1. Amazon Bedrock
   - Dokumentation: https://aws.amazon.com/bedrock/
   - Unterstützte Modelle:
     - Amazon
       - Nova Lite
       - Nova Micro
       - Nova Pro
     - AI21 Labore
       - Jamba 1.5 Mini
       - Jamba 1.5 Large
     - Anthropic
       - Claude v3 Haiku
       - Claude v3.5 Sonett
       - Claude v3.7 Sonnet (mithilfe von Inferenzprofilen)
     - Cohere
       - Command R
       - Command R+
     - Deepseek
       - Deepseek-R1 (durch die Verwendung von Inferenzprofilen)
     - Meta
       - Llama 3
       - Llama 3.2 (durch die Verwendung von Inferenzprofilen)
     - Mistral AI

        - Mistral 7B Instruct
        - Mistral 8x7B Instruktor
    - Regionsübergreifende Inferenz
        - Möglichkeit, Inferenzprofile zu verwenden, die in derselben Region wie das Deployment-Dashboard definiert sind
2. Amazon SageMaker KI
    - Dokumentation: https://aws.amazon.com/sagemaker/
    - Unterstützte Modelle: Text-to-Text-Modelle

Die neuesten Modellparameter, bewährten Methoden und Anwendungsempfehlungen finden Sie in der Dokumentation der Modellanbieter.

Datenerfassung

Diese Lösung sendet Betriebsmetriken (die „Daten“) über die Verwendung dieser Lösung an AWS. Wir verwenden diese Daten, um besser zu verstehen, wie Kunden diese Lösung und die damit verbundenen Dienstleistungen und Produkte nutzen. Die Erfassung dieser Daten durch AWS unterliegt der AWS-Datenschutzerklärung.

Mitwirkende

- Tarek Abdunabi
- Majd Arbash
- George Bearden
- Mukit bin Momin
- Michael Connor
- Johnny Duval
- Nihit Kasabwala
- Ahern Knox
- Simon Kröll
- Michael Lin
- Tim Mekari

- Ibrahim Mohamed
- Omar Radwan Mohsen
- James Nixon
- Dekshitha Ravikumar
- Jae Shim
- Ajay Swamy
- Mohamed Taha
- Reet Takkar
- Dmitri Tschikatilow
- Jason Wreath
- Kamyar Ziabari

Überarbeitungen

Veröffentlichungsdatum: Oktober 2023 (letzte Aktualisierung: Januar 2025)

In der Datei CHANGELOG.md im GitHub Repository finden Sie alle wichtigen Änderungen und Aktualisierungen der Software. Das Changelog enthält eine übersichtliche Aufzeichnung der Verbesserungen und Korrekturen für jede Version.

Hinweise

Kunden sind dafür verantwortlich, Ihre eigene unabhängige Bewertung der Informationen in diesem Dokument vorzunehmen. Dieses Dokument: (a) dient nur zu Informationszwecken, (b) stellt aktuelle Produktangebote und Praktiken von AWS dar, die ohne vorherige Ankündigung geändert werden können, und (c) stellt keine Verpflichtungen oder Zusicherungen von AWS und seinen verbundenen Unternehmen, Lieferanten oder Lizenzgebern dar. AWS-Produkte oder -Services werden „wie sie sind“ ohne ausdrückliche oder stillschweigende Garantien, Zusicherungen oder Bedingungen jeglicher Art bereitgestellt. Die Verantwortlichkeiten und Verbindlichkeiten von AWS gegenüber seinen Kunden werden durch AWS-Verträge geregelt, und dieses Dokument ist weder Teil einer Vereinbarung zwischen AWS und seinen Kunden noch ändert es diese.

Generative AI Application Builder auf AWS ist unter den Bedingungen der Apache License Version 2.0 lizenziert.

Important

Generative AI Application Builder auf AWS ermöglicht es Ihnen, generative Anwendungen für künstliche Intelligenz auf AWS zu erstellen und bereitzustellen, indem Sie das generative KI-Modell Ihrer Wahl nutzen, einschließlich generativer KI-Modelle von Drittanbietern, die Sie verwenden können und die nicht Eigentum von AWS sind oder über die AWS keine Kontrolle hat („Generative KI-Modelle von Drittanbietern“).
Ihre Nutzung der generativen KI-Modelle von Drittanbietern unterliegt den Bedingungen, die Ihnen von den Drittanbietern generativer KI-Modelle zur Verfügung gestellt wurden, als Sie Ihre Lizenz für deren Nutzung erworben haben (z. B. deren Nutzungsbedingungen, Lizenzvereinbarung, Nutzungsbedingungen und Datenschutzrichtlinie).
Sie sind dafür verantwortlich, sicherzustellen, dass Ihre Nutzung der generativen KI-Modelle von Drittanbietern den für sie geltenden Bedingungen sowie allen für Sie geltenden Gesetzen, Regeln, Vorschriften, Richtlinien oder Standards entspricht.
Sie sind auch dafür verantwortlich, Ihre eigene unabhängige Bewertung der von Ihnen verwendeten generativen KI-Modelle von Drittanbietern vorzunehmen, einschließlich ihrer Ergebnisse und der Art und Weise, wie Drittanbieter generativer KI-Modelle alle Daten verwenden, die aufgrund Ihrer Bereitstellung an sie übertragen werden könnten. AWS gibt keine Zusicherungen, Gewährleistungen oder Garantien in Bezug auf die generativen KI-Modelle von Drittanbietern ab, bei denen es sich im Rahmen Ihrer Vereinbarung mit AWS um

„Inhalte von Drittanbietern“ handelt. Generative AI Application Builder auf AWS wird Ihnen im Rahmen Ihrer Vereinbarung mit AWS als „AWS-Inhalt“ angeboten.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.