Guida all'implementazione

# Generative AI Application Builder su AWS

# Generative AI Application Builder su AWS: Guida all'implementazione

# Table of Contents

# Questa soluzione facilita lo sviluppo, la sperimentazione rapida e l'implementazione di applicazioni di intelligenza artificiale generativa (AI)

Generative AI Application Builder su AWS facilita lo sviluppo, la sperimentazione rapida e la distribuzione di applicazioni di intelligenza artificiale generativa (AI) senza richiedere una profonda esperienza nell'IA. Questa soluzione AWS accelera lo sviluppo e semplifica la sperimentazione aiutandoti a:

- Inserisci dati e documenti specifici della tua azienda
- Valuta e confronta le prestazioni di modelli linguistici di grandi dimensioni () LLMs
- Esegui attività e flussi di lavoro in più fasi con agenti AI
- Crea rapidamente applicazioni estensibili e implementale con un'architettura di livello aziendale

Generative AI Application Builder su AWS include integrazioni con:

- LLMs disponibile su Amazon Bedrock
- LLMs che hai distribuito su Amazon SageMaker AI
- Basi di conoscenza di Amazon Bedrock per la Retrieval-Augmented Generation (RAG)
- Amazon Bedrock Guardrails implementerà misure di sicurezza e ridurrà le allucinazioni
- Amazon Bedrock Agents per creare flussi di lavoro agentici in grado di eseguire l'orchestrazione e il completamento delle attività
- Amazon Bedrock AgentCore per creare, implementare e gestire agenti AI pronti per la produzione con supporto di runtime esteso
- Server Model Context Protocol (MCP) per l'integrazione di dati e strumenti aziendali

Inoltre, questa soluzione consente le connessioni al modello prescelto utilizzando LangChain connettori. Questi connettori sono disponibili in una funzione AWS Lambda che viene distribuita con la soluzione. Puoi iniziare con la procedura guidata di implementazione senza codice per creare applicazioni AI generative per la ricerca conversazionale, i chatbot generati dall'intelligenza artificiale, la generazione di testo e il riepilogo del testo.

Questa guida all'implementazione fornisce una panoramica della soluzione Generative AI Application Builder on AWS, della sua architettura e dei suoi componenti di riferimento, considerazioni per la pianificazione della distribuzione e i passaggi di configurazione per la distribuzione della soluzione nel cloud Amazon Web Services (AWS).

Questa guida è destinata agli architetti di soluzioni, ai responsabili delle decisioni aziendali, DevOps agli ingegneri, ai data scientist e ai professionisti del cloud che desiderano implementare Generative AI Application Builder su AWS nel proprio ambiente.

Utilizza questa tabella di navigazione per trovare rapidamente le risposte a queste domande:

| Se vuoi. | Leggere.. |
| --- | --- |
| Conosci il costo di esecuzione di questa soluzione.<br><br>Il costo stimato per l'esecuzione di questa soluzione varia in base ai componenti distribuiti e al numero di query.<br><br>Il costo per eseguire la dashboard di Deployment con parametri predefiniti e 100 utenti attivi nella regione Stati Uniti orientali (Virginia settentrionale) per un mese è di circa 20,12 USD al mese.<br><br>Il costo di un caso d'uso Text implementato senza RAG per 1 utente aziendale che esegue 100 query al giorno con LLM è di circa 12,39 USD al mese.<br><br>Il costo di un use case abilitato a RAG con un indice Amazon Kendra che supporta 8.000 interazioni al giorno è di circa 204,26 USD al mese, più il costo della knowledge base. | Costo |
| Comprendi le considerazioni sulla sicurezza relative a questa soluzione. | Sicurezza |

| Se vuoi. | Leggere.. |
| --- | --- |
| Scopri come pianificare le quote per questa soluzione. | Quote |
| Scopri quali regioni AWS supportano questa soluzione. | Regioni AWS supportate |
| Visualizza o scarica il CloudFormation modello AWS incluso in questa soluzione per distribuire automaticamente le risorse dell'infrastruttura (lo «stack») per questa soluzione. | CloudFormation Modello AWS |
| Accedi al codice sorgente e, facoltativamente, utilizza AWS Cloud Development Kit (AWS CDK) per distribuire la soluzione. | GitHub repository |

# Funzionalità e vantaggi

La soluzione Generative AI Application Builder on AWS offre le seguenti funzionalità:

Sperimentazione rapida

Questa soluzione consente agli utenti di sperimentare rapidamente eliminando il carico di lavoro necessario per implementare più istanze con configurazioni diverse e confrontare output e prestazioni. Sperimenta configurazioni multiple di vari parametri LLMs, tra cui progettazione tempestiva, knowledge base aziendali, guardrail, agenti di intelligenza artificiale e altri parametri.

Scelta e configurabilità

Con connettori predefiniti per una varietà di modelli LLMs, ad esempio disponibili tramite Amazon Bedrock, questa soluzione ti offre la flessibilità necessaria per implementare il modello che preferisci, oltre ai servizi AWS e ai principali servizi FM che preferisci. Puoi anche consentire ad Amazon Bedrock Agents di svolgere varie attività e flussi di lavoro.

Agente Builder

Crea e distribuisci agenti AI pronti per la produzione con una gestione completa del ciclo di vita. Configura le istruzioni di sistema, integra i server Model Context Protocol (MCP) per gli strumenti

aziendali e l'accesso ai dati e abilita le funzionalità di memoria per la conservazione del contesto tra le conversazioni. Gli agenti vengono distribuiti su Amazon Bedrock AgentCore con supporto di runtime esteso e risposte di streaming in tempo reale.

Workflow Builder

Orchestra più agenti Agent Builder in flussi di lavoro complessi utilizzando la delega gerarchica. Crea un agente supervisore che selezioni e coordini in modo autonomo agenti Agent Builder specializzati per gestire attività in più fasi. Configura le descrizioni degli agenti, le strategie di delega e la memoria a livello di flusso di lavoro riutilizzando le distribuzioni esistenti di Agent Builder.

Pronto per la produzione

Costruita secondo i principi di progettazione di AWS Well-Architected, questa soluzione offre sicurezza e scalabilità di livello aziendale con elevata disponibilità e bassa latenza, garantendo una perfetta integrazione nelle applicazioni con standard di prestazioni elevati.

Architettura modulare estensibile

Estendi le funzionalità di questa soluzione integrando i tuoi progetti esistenti o connettendo nativamente servizi AWS aggiuntivi. Poiché si tratta di un'applicazione open source, puoi utilizzare il livello di LangChain orchestrazione incluso o le funzioni Lambda per connetterti ai servizi di tua scelta.

Integrazione con Service Catalog AppRegistry e Application Manager, una funzionalità di AWS Systems Manager

Questa soluzione include una AppRegistry risorsa Service Catalog per registrare il CloudFormation modello della soluzione e le relative risorse sottostanti come applicazione sia in AWS Service Catalog AppRegistry che in AWS Systems Manager Application Manager. Con questa integrazione, puoi gestire centralmente le risorse della soluzione.

# Caso d'uso tra Agent Builder e Bedrock Agent

Questa soluzione offre due approcci distinti per lavorare con gli agenti AI, ciascuno adatto a diversi casi d'uso e requisiti:

| Funzionalità | Caso d'uso di Bedrock Agent | Agente Builder |
|---|---|---|
| Scopo | Richiama agenti Amazon Bedrock predistribuiti | Crea, distribuisci e gestisci agenti personalizzati |
| Configurazione | Solo ID agente e ID alias | Configurazione completa dell'agente: istruzioni di sistema, modelli, server MCP, memoria |
| Distribuzione | Livello di invocazione semplice | Ciclo di vita completo dell'agente su Runtime AgentCore |
| Runtime | Servizio Amazon Bedrock Agents | Amazon Bedrock AgentCore con SDK Strands |
| Integrazione degli strumenti | Configurato nella console Bedrock Agents | Server Model Context Protocol (MCP) e strumenti Strands integrati |
| Memoria | Gestito da Bedrock Agents (fino a 30 giorni) | AgentCore Memoria con conservazione configurabile a breve e lungo termine |
| Personalizzazione | Limitato alle impostazioni degli agenti predistribuite | Controllo completo su istruzioni, modelli, strumenti e comportamenti |
| Ideale per | Implementazione rapida degli agenti esistenti | Implementazioni personalizzate per lo sviluppo e la produzione di agenti |

> **Note**
>
> Entrambe le opzioni supportano lo streaming in tempo reale, la cronologia delle conversazioni e la sicurezza di livello aziendale.

## Workflow Builder

Workflow Builder consente l'orchestrazione di più agenti creando un agente supervisore che delega il lavoro ad agenti Agent Builder specializzati. Ogni flusso di lavoro è composto da:

- Agente supervisore: l'agente entrypoint che riceve le richieste degli utenti e coordina gli agenti specializzati
- Agenti specializzati: Agent Builder utilizza casi a cui il supervisore può delegare attività
- Modello Agent as Tools: il supervisore registra ogni agente Agent Builder come strumento e seleziona autonomamente gli agenti da utilizzare

| Funzionalità | Agent Builder | Generatore di flussi di lavoro |
|---|---|---|
| Scopo | Crea e distribuisci singoli agenti personalizzati | Orchestra più agenti Agent Builder |
| Tipo di agente | Agente singolo con strumenti MCP | Agente supervisore e più agenti Agent Builder |
| Integrazione degli strumenti | Server MCP e strumenti Strands | Agenti Agent Builder registrati come strumenti |
| Delega | Invocazione diretta dello strumento | Selezione e delega autonome degli agenti |
| Complessità | Attività con un solo agente | Flussi di lavoro in più fasi e con più agenti |
| Riutilizzo degli agenti | N/D | Riutilizza le distribuzioni esistenti di Agent Builder |
| Ideale per | Attività mirate e a dominio singolo | Flussi di lavoro complessi che richiedono specializzazioni multiple |

**Note**

- I flussi di lavoro richiedono almeno un caso d'uso di Agent Builder come agente specializzato
- Tutti gli agenti specializzati devono essere casi d'uso di Agent Builder distribuiti in GAAB

# Casi d’uso

Risposta alle domande tramite dati aziendali

LLMs e altri modelli di base sono stati pre-addestrati su un ampio corpus di dati, che consente loro di svolgere bene molte attività di elaborazione del linguaggio naturale (NLP). Ma la maggior parte dei modelli di base LLMs sono statici e sono stati preformati, il che limita la loro capacità di rispondere con precisione a domande su argomenti nuovi, specializzati o proprietari. Utilizzando l'apprendimento basato sulla richiesta, è possibile sfruttare le potenti funzionalità di NLP e generazione di testi di un LLM per fornire ai clienti esperienze più complete sulla base dei dati aziendali.

Prototipazione rapida di intelligenza artificiale generativa

Pronta all'uso, la soluzione viene fornita in bundle con vari fornitori di modelli e casi d'uso. Con una procedura guidata di implementazione facile da usare, i clienti possono implementare casi d'uso predefiniti per consentire la rapida sperimentazione di diversi prototipi e carichi di lavoro di intelligenza artificiale generativa.

Confronto e sperimentazione di più LLM

LLMs funzionano in modo diverso e, date le esigenze specifiche della vostra applicazione, potreste scoprire che un LLM si adatta meglio alla vostra applicazione rispetto a un altro. Ciò può essere dovuto a ragioni legate alle prestazioni, alla precisione, ai costi, alla creatività o a molti altri fattori. Questa soluzione consente di implementare rapidamente più casi d'uso, consentendoti di sperimentare e confrontare diverse configurazioni fino a trovare quella che soddisfa le tue esigenze.

# Concetti e definizioni

Questa sezione descrive i concetti chiave e definisce la terminologia specifica di questa soluzione:

utente amministratore

Nel contesto di questa guida, l'utente amministratore è l'unico responsabile della gestione dei contenuti contenuti nella distribuzione. Questo utente ha accesso all'interfaccia utente del dashboard di Deployment ed è il principale responsabile della cura dell'esperienza utente aziendale. Questo è il nostro cliente target principale.

utente aziendale

Nel contesto di questa guida, l'utente aziendale rappresenta le persone per le quali è stato implementato lo use case. Sono gli utenti della knowledge base e il cliente responsabile della valutazione e della sperimentazione di. LLMs

Dashboard di implementazione

La dashboard di distribuzione è un'interfaccia web che funge da console di gestione per consentire agli utenti amministratori di visualizzare, gestire e creare i propri casi d'uso. Questa dashboard consente ai clienti di sperimentare, iterare e mettere in produzione rapidamente vari AI/ML carichi di lavoro sfruttando i vantaggi. LLMs

DevOps utente

Nel contesto di questa guida, l' DevOps utente è responsabile della distribuzione della soluzione all'interno dell'account AWS e della gestione dell'infrastruttura, dell'aggiornamento della soluzione, del monitoraggio delle prestazioni e del mantenimento dello stato generale e del ciclo di vita della soluzione.

caso d'uso

I casi d'uso sono applicazioni isolate dalla soluzione complessiva che si integrano LLMs per consentire ai clienti esperienze più complete grazie all'aggiunta di un'interfaccia in linguaggio naturale in applicazioni nuove o esistenti. I casi d'uso possono essere implementati tramite la dashboard di Deployment o da soli.

> **Note**
>
> Per un riferimento generale ai termini di AWS, consulta il Glossario AWS.

# Panoramica dell’architettura

Questa sezione fornisce diagrammi di architettura di implementazione di riferimento per i componenti distribuiti con questa soluzione.

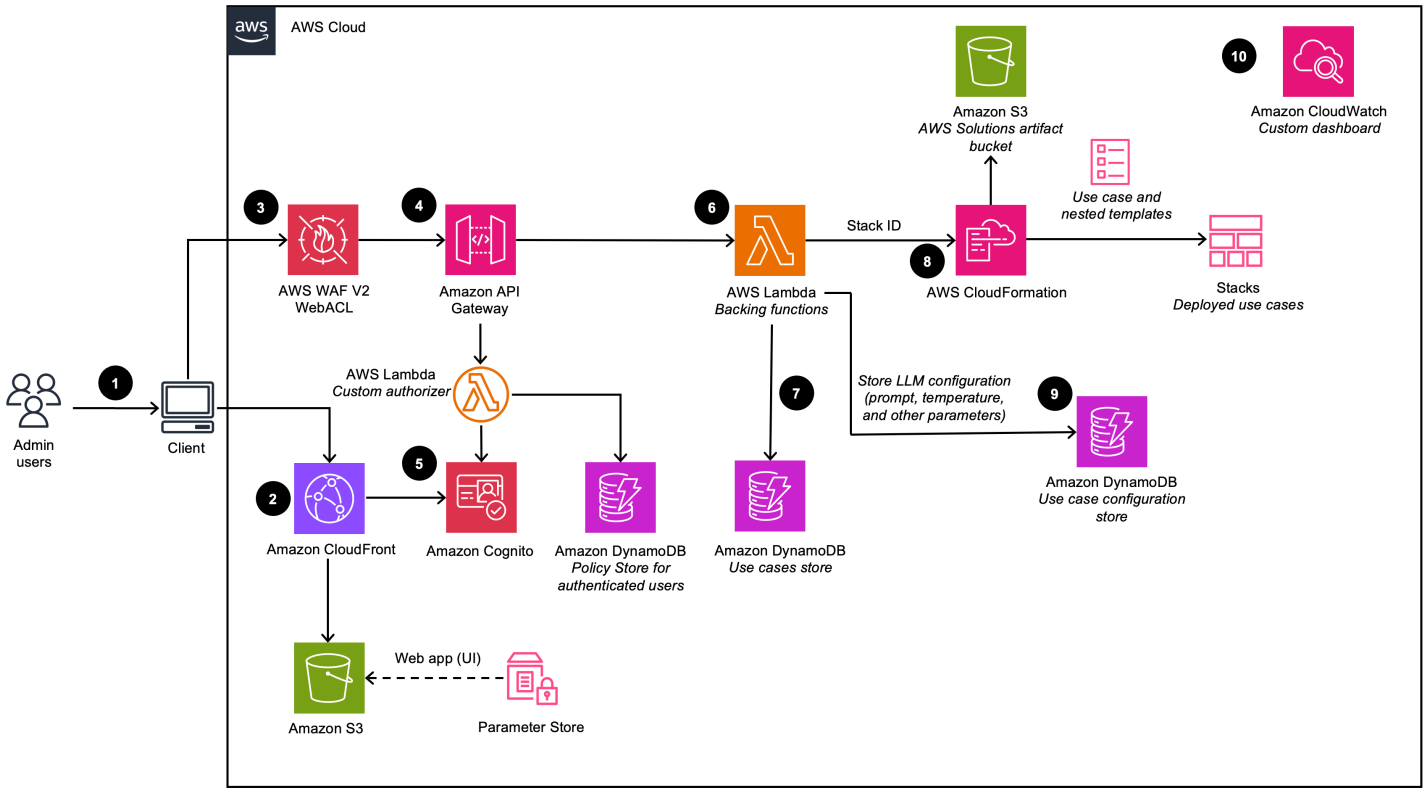
## Diagrammi di architettura

Per supportare diversi casi d'uso ed esigenze aziendali, questa soluzione fornisce sei CloudFormation modelli AWS:

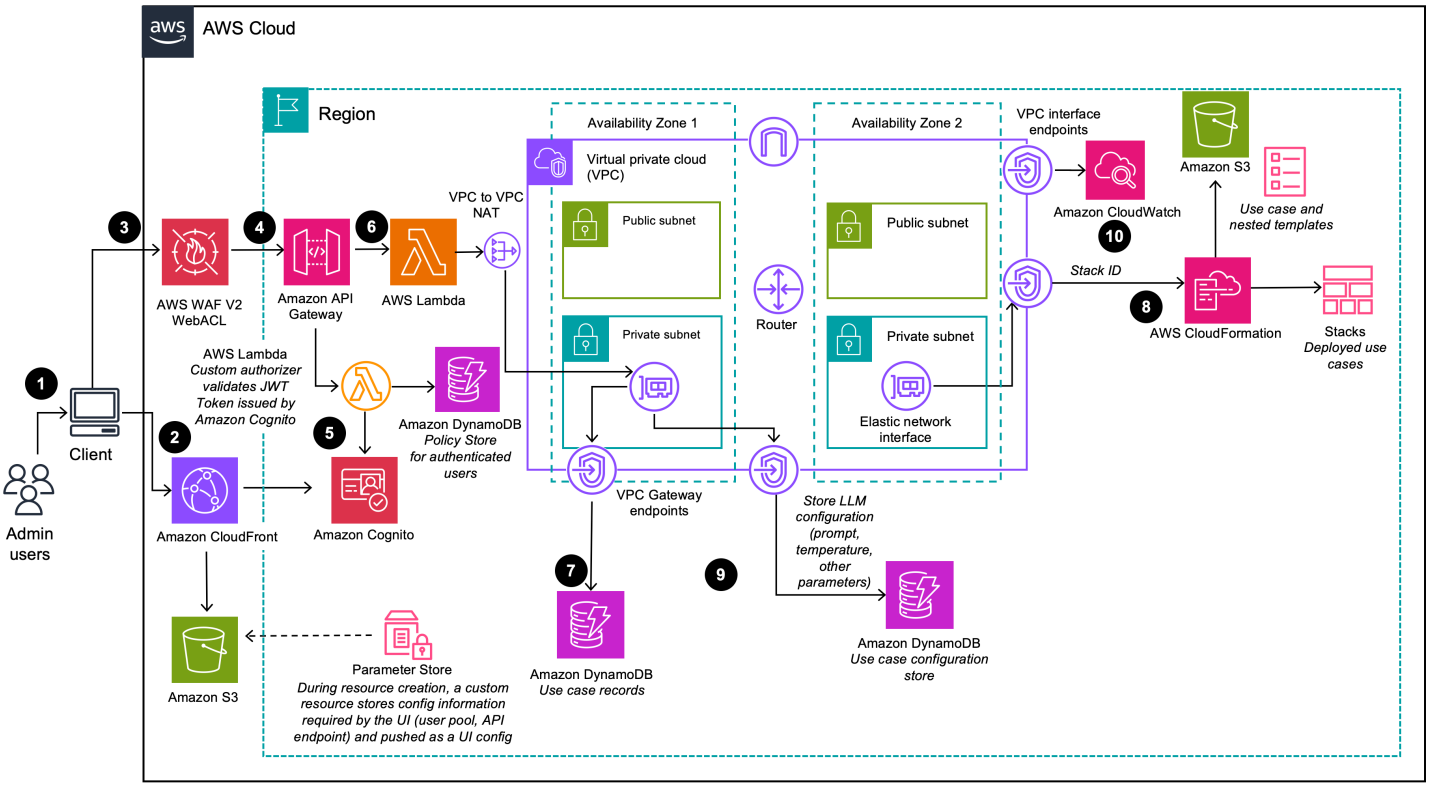
1. Dashboard di distribuzione: la dashboard di distribuzione è un'interfaccia Web che funge da console di gestione per consentire agli utenti amministratori di visualizzare, gestire e creare i propri casi d'uso. Questa dashboard consente ai clienti di sperimentare, iterare e produrre rapidamente vari AI/ML carichi di lavoro sfruttando i vantaggi. LLMs
2. Caso d'uso testuale: lo use case Text consente agli utenti di sperimentare un'interfaccia in linguaggio naturale utilizzando l'intelligenza artificiale generativa. Questo caso d'uso può essere integrato in applicazioni nuove o esistenti ed è implementabile tramite la dashboard di distribuzione o indipendentemente tramite un URL fornito.
3. Caso d'uso di Bedrock Agent: lo use case Bedrock Agent consente l'uso di agenti Bedrock esistenti per completare attività o automatizzare flussi di lavoro ripetuti.
4. Server MCP: lo use case MCP Server consente l'implementazione e la gestione di server Model Context Protocol che forniscono un accesso standardizzato a strumenti e risorse alle applicazioni AI. Supporta sia i metodi gateway per il wrapping delle funzioni Lambda esistenti, sia i server MCP esterni APIs, sia i metodi di runtime per l'implementazione di server MCP containerizzati personalizzati.
5. Agent Builder: Agent Builder consente la creazione e l'implementazione di agenti AI pronti per la produzione su Amazon Bedrock AgentCore con controllo completo della configurazione, integrazione del server MCP e funzionalità di gestione della memoria.
6. Workflow Builder - Workflow Builder consente la creazione di agenti supervisori che orchestrano più agenti Agent Builder utilizzando il modello di delega Agents as Tools per flussi di lavoro complessi con più agenti.

### Dashboard di implementazione

Descrive l'architettura del dashboard di distribuzione (se distribuito con l'opzione VPC disabilitata)




## Descrive l'architettura del dashboard di distribuzione (se distribuito con l'opzione VPC abilitata)



> **Note**
>
> Le CloudFormation risorse AWS vengono create a partire da costrutti di AWS Cloud Development Kit (AWS CDK).

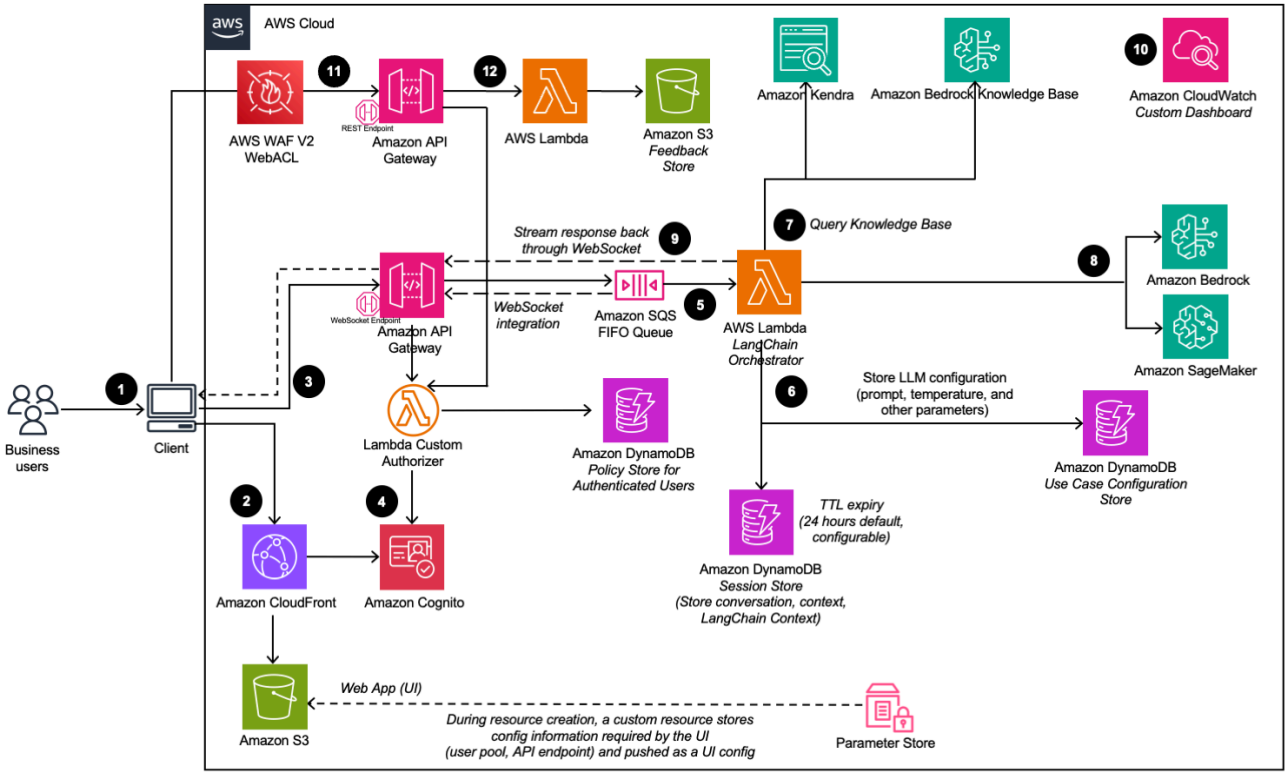
Il flusso di processo di alto livello per i componenti della soluzione distribuiti con il CloudFormation modello AWS è il seguente:

1. Gli utenti amministratori accedono all'interfaccia utente (UI) di Deployment Dashboard.
2. Amazon CloudFront offre l'interfaccia utente Web, ospitata in un bucket Amazon Simple Storage Service (Amazon S3).
3. AWS WAF li protegge APIs dagli attacchi. Questa soluzione configura una serie di regole denominate Web Access Control List (Web ACL) che consente, blocca o conta le richieste Web in base a regole e condizioni di sicurezza Web configurabili e definite dall'utente.
4. L'interfaccia utente Web sfrutta un set di REST APIs che vengono esposti utilizzando Amazon API Gateway.
5. Amazon Cognito autentica gli utenti e supporta sia l'interfaccia utente CloudFront Web che l'API Gateway.
6. AWS Lambda fornisce la logica di business per gli endpoint REST. Questa funzione Lambda di supporto gestisce e crea le risorse necessarie per eseguire implementazioni di use case utilizzando AWS. CloudFormation
7. Amazon DynamoDB archivia l'elenco delle distribuzioni.
8. Quando un nuovo caso d'uso viene creato dall'utente amministratore, la funzione Lambda di backup avvia CloudFormation un evento di creazione dello stack per il caso d'uso richiesto.
9. Tutte le opzioni di configurazione LLM fornite dall'utente amministratore nella procedura guidata di distribuzione vengono salvate in DynamoDB. La distribuzione utilizza questa tabella DynamoDB per configurare l'LLM in fase di esecuzione.
10. Utilizzando Amazon CloudWatch, questa soluzione raccoglie metriche operative da vari servizi per generare dashboard personalizzati che consentono di monitorare le prestazioni e lo stato operativo della soluzione.

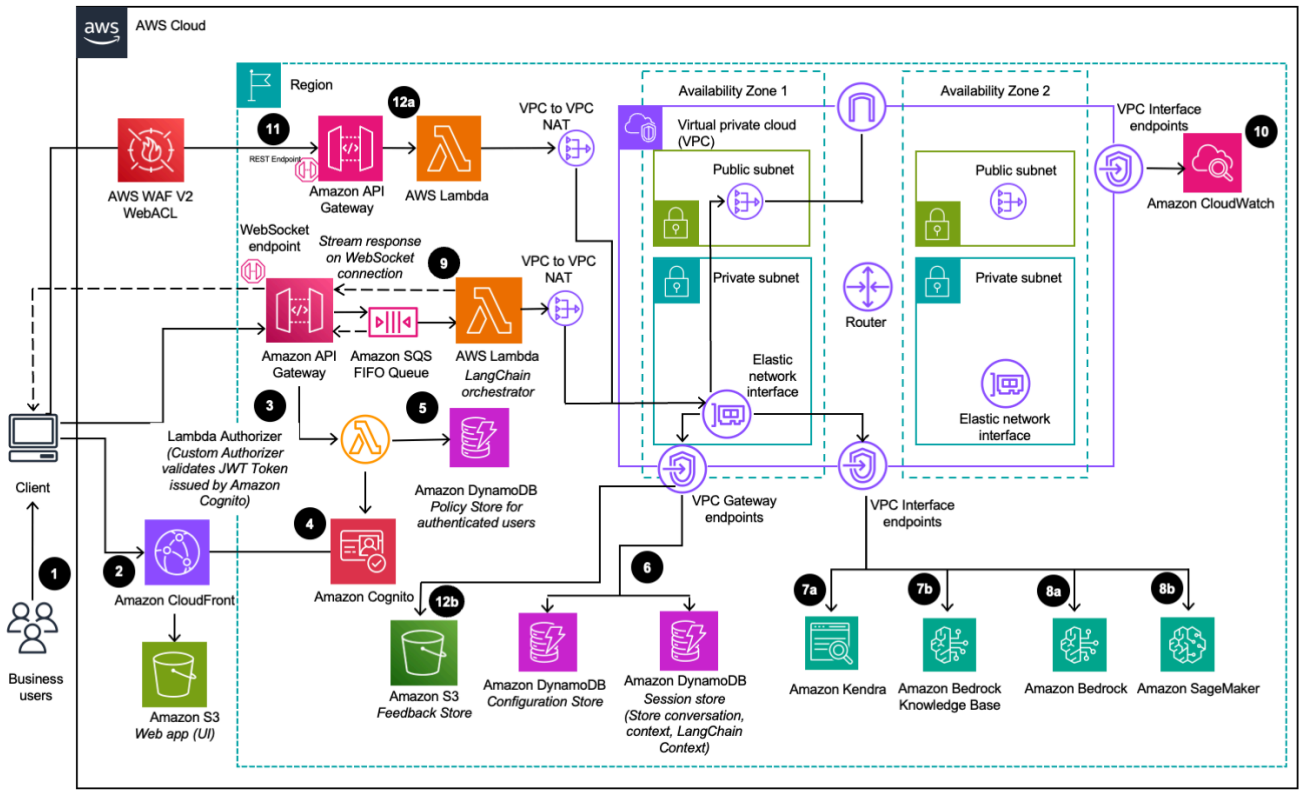
> **Note**
>
> - Se scegli di implementare questa soluzione in un Amazon VPC, i dati verranno instradati all'interno della tua rete privata.
> - Sebbene la dashboard di distribuzione possa essere avviata nella maggior parte delle regioni AWS, i casi d'uso implementati presentano alcune restrizioni in base alla disponibilità del servizio. Per ulteriori dettagli, consulta le regioni AWS supportate.

## Caso di utilizzo del testo

Descrive l'architettura dei casi d'uso di testo (se distribuito con l'opzione VPC disabilitata)




Descrive l'architettura dei casi d'uso di testo (se distribuito con l'opzione VPC abilitata)




Il flusso di processo di alto livello per i componenti della soluzione distribuiti con il CloudFormation modello AWS è il seguente:

1. Gli utenti amministratori distribuiscono lo use case utilizzando il Deployment Dashboard. Gli utenti aziendali accedono all'interfaccia utente dello use case.
2. CloudFront fornisce l'interfaccia utente Web ospitata in un bucket S3.
3. L'interfaccia utente Web sfrutta un' WebSocket integrazione creata utilizzando API Gateway. L'API Gateway è supportato da una funzione di autorizzazione Lambda personalizzata, che restituisce la policy AWS Identity and Access Management (IAM) appropriata basata sul gruppo Amazon Cognito a cui appartiene l'utente che effettua l'autenticazione. La policy è archiviata in DynamoDB.
4. Amazon Cognito autentica gli utenti e supporta sia l'interfaccia utente CloudFront Web che l'API Gateway.
5. Le richieste in arrivo dall'utente aziendale vengono passate da API Gateway a una coda Amazon SQS e quindi a Orchestrator. LangChain  L'LangChain Orchestrator è una raccolta di funzioni e livelli Lambda che forniscono la logica aziendale per soddisfare le richieste provenienti dall'utente aziendale. La coda consente il funzionamento asincrono dell'integrazione tra API Gateway e Lambda. La coda passa le informazioni di connessione alle funzioni Lambda che invieranno

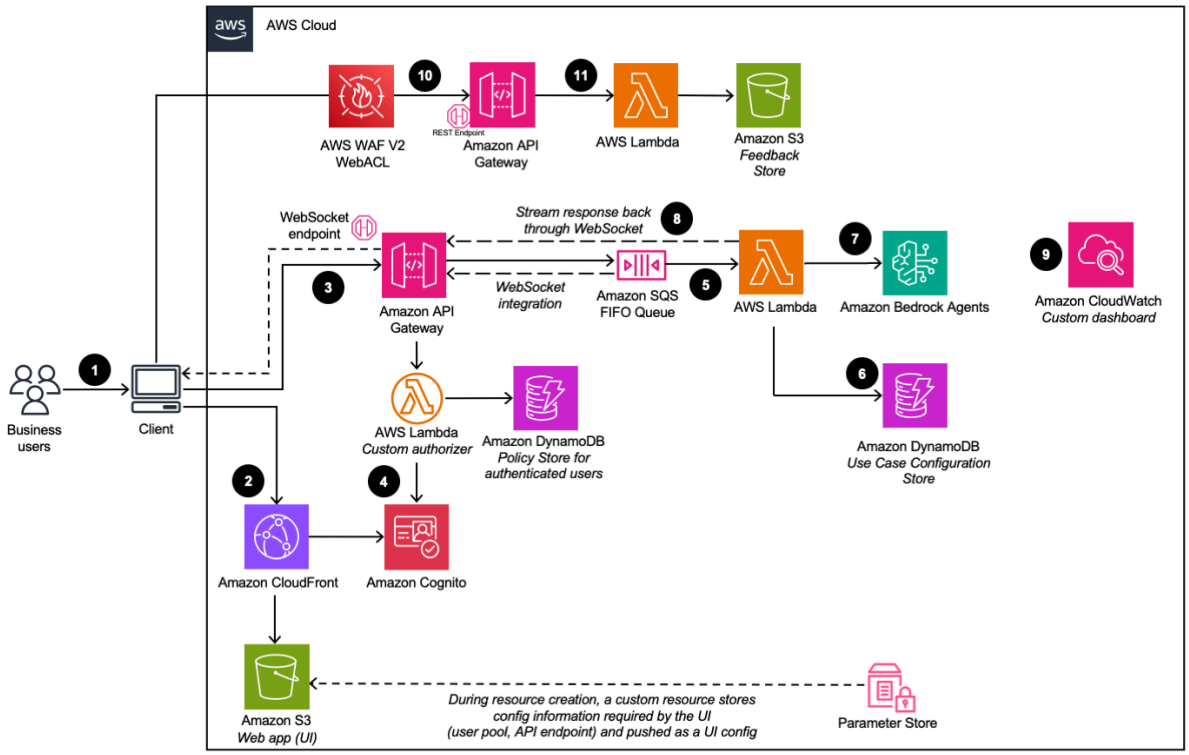
quindi i risultati direttamente alla connessione websocket API Gateway per supportare chiamate di inferenza a lunga durata.

6. L'LangChain Orchestrator utilizza Amazon DynamoDB per ottenere le opzioni LLM configurate e le informazioni necessarie sulla sessione (come la cronologia chat).
7. Se la distribuzione ha una knowledge base abilitata, LangChain Orchestrator sfrutta Amazon Kendra o Knowledge Bases for Amazon Bedrock per eseguire una query di ricerca per recuperare estratti di documenti.
8. Utilizzando la cronologia della chat, la query e il contesto della knowledge base, LangChain Orchestrator crea il prompt finale e invia la richiesta al LLM ospitato su Amazon Bedrock o Amazon AI. SageMaker
9. Quando la risposta ritorna dal LLM, l'LangChain Orchestrator trasmette la risposta attraverso l'API Gateway WebSocket per essere utilizzata dall'applicazione client.
10. Utilizzando Amazon CloudWatch, questa soluzione raccoglie metriche operative da vari servizi per generare dashboard personalizzati che consentono di monitorare le prestazioni e lo stato operativo della distribuzione.
11. Se la raccolta di feedback è abilitata, viene reso disponibile un endpoint API REST che sfrutta Amazon API Gateway per la raccolta del feedback degli utenti.
12. Il feedback a supporto di lambda, amplia il feedback inviato con metadati aggiuntivi specifici del caso d'uso (ad esempio il modello utilizzato) e archivia i dati in Amazon S3 per analisi e report successivi da parte degli utenti. DevOps

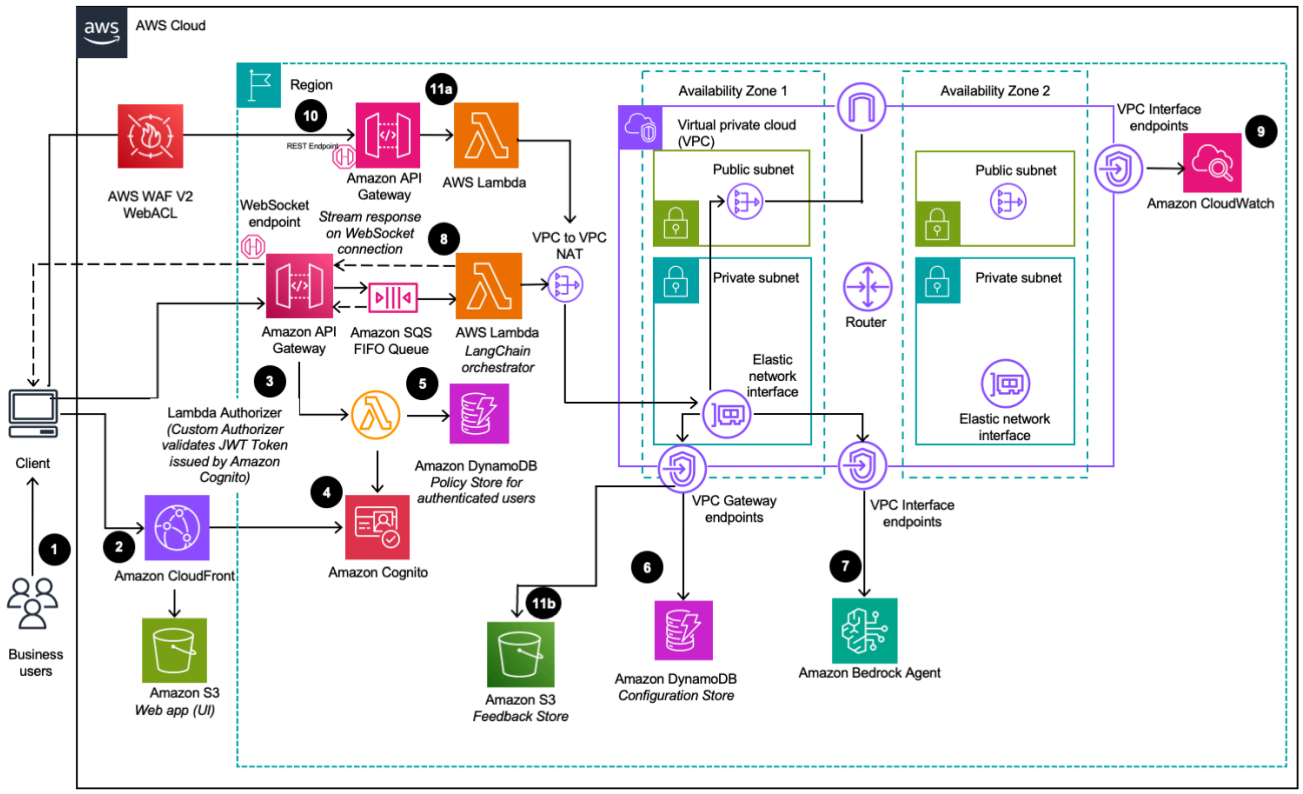
> **Note**
>
> Se scegli di implementare questa soluzione in un Amazon VPC, i dati verranno indirizzati alla tua rete privata.

## Caso d'uso di Bedrock Agent

Descrive l'architettura degli use case di Bedrock Agent (se distribuito con l'opzione VPC disabilitata)




Descrive l'architettura degli use case di Bedrock Agent (se distribuito con l'opzione VPC abilitata)



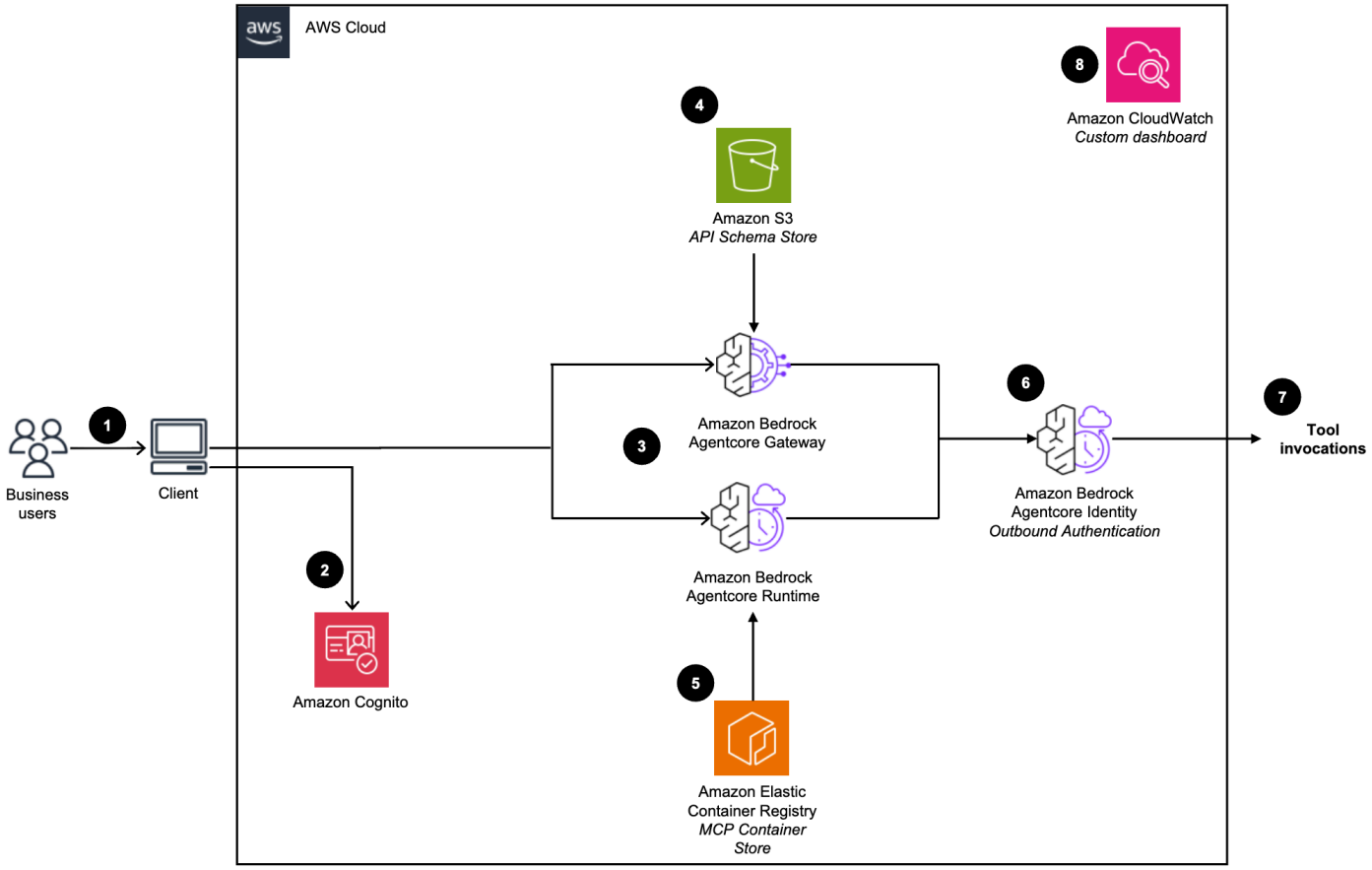
Il flusso di processo di alto livello per i componenti della soluzione distribuiti con il CloudFormation modello AWS è il seguente:

1. Gli utenti amministratori distribuiscono lo use case utilizzando il Deployment Dashboard. Gli utenti aziendali accedono all'interfaccia utente dello use case.
2. CloudFront fornisce l'interfaccia utente Web ospitata in un bucket S3.
3. L'interfaccia utente Web sfrutta un' WebSocket integrazione creata utilizzando API Gateway. L'API Gateway è supportato da una funzione di autorizzazione Lambda personalizzata, che restituisce la policy AWS Identity and Access Management (IAM) appropriata basata sul gruppo Amazon Cognito a cui appartiene l'utente che effettua l'autenticazione. La policy è archiviata in DynamoDB.
4. Amazon Cognito autentica gli utenti e supporta sia l'interfaccia utente CloudFront Web che l'API Gateway.
5. Le richieste in arrivo dall'utente aziendale vengono passate da API Gateway a una coda Amazon SQS e quindi alla funzione AWS Lambda. La coda consente il funzionamento asincrono dell'integrazione tra API Gateway e Lambda. La coda passa le informazioni di connessione alla funzione Lambda che invierà quindi i risultati direttamente alla connessione websocket API Gateway per supportare chiamate di inferenza a lunga durata.
6. La funzione AWS Lambda utilizza Amazon DynamoDB per ottenere le configurazioni dei casi d'uso in base alle esigenze.
7. Utilizzando l'input dell'utente e tutte le configurazioni dei casi d'uso pertinenti, la funzione AWS Lambda crea e invia un payload di richiesta all'agente Amazon Bedrock configurato per soddisfare l'intento dell'utente.
8. Quando la risposta arriva dall'Amazon Bedrock Agent, la funzione Lambda trasmette la risposta attraverso l'API WebSocket Gateway per essere utilizzata dall'applicazione client.
9. Utilizzando Amazon CloudWatch, questa soluzione raccoglie metriche operative da vari servizi per generare dashboard personalizzati che consentono di monitorare le prestazioni e lo stato operativo della distribuzione.
10. Se la raccolta di feedback è abilitata, viene reso disponibile un endpoint API REST che sfrutta Amazon API Gateway per la raccolta del feedback degli utenti.
11. Il feedback a supporto di lambda, amplia il feedback inviato con metadati aggiuntivi specifici dei casi d'uso e archivia i dati in Amazon S3 per analisi e report successivi da parte degli utenti. DevOps

> **Note**
>
> Se scegli di implementare questa soluzione in un Amazon VPC, i dati verranno instradati all'interno della tua rete privata.

## Caso d'uso del server MCP

Descrive l'architettura dei casi d'uso del server MCP




Lo use case MCP Server consente l'implementazione e la gestione di server Model Context Protocol su Amazon AgentCore Bedrock. I server MCP forniscono un'interfaccia standardizzata per le applicazioni AI per accedere a strumenti, risorse e fonti di dati aziendali.

La soluzione supporta due metodi di implementazione:

- Metodo gateway: racchiude funzioni Lambda esistenti, REST o server MCP esterni come strumenti MCP APIs, gestendo automaticamente la traduzione del protocollo

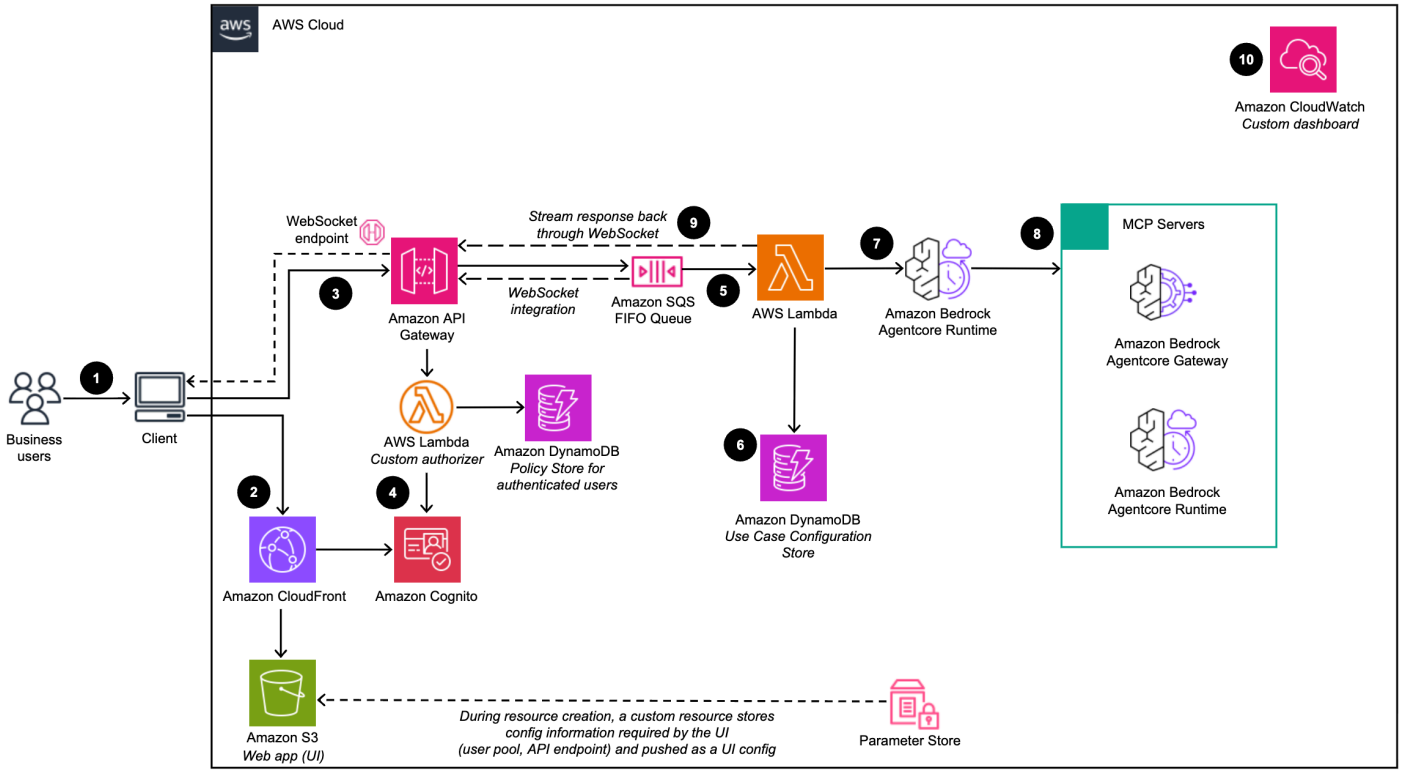
- Metodo di runtime: distribuisce server MCP containerizzati personalizzati da immagini Amazon ECR

Il flusso di processo di alto livello per la distribuzione di MCP Server è il seguente:

1. Gli utenti amministratori distribuiscono lo use case MCP Server utilizzando il Deployment Dashboard, selezionando il metodo di distribuzione Gateway o Runtime.
2. Questa azione è autenticata con Amazon Cognito.
3. Per l'implementazione del Gateway, la soluzione crea un Amazon Bedrock AgentCore Gateway che trasforma le funzioni Lambda esistenti o i server MCP esterni in strumenti conformi a MCP. APIs Per la distribuzione Runtime, la soluzione distribuisce server MCP containerizzati su Amazon Bedrock AgentCore Runtime utilizzando immagini ECR fornite.
4. Le implementazioni gateway recuperano gli API/Lambda/Smithy schemi necessari dalla posizione di caricamento in Amazon S3 o si connettono direttamente agli endpoint URL del server MCP.
5. Le distribuzioni di runtime recuperano il server MCP containerizzato fornito dall'utente da Amazon Elastic Container Registry (ECR)
6. Il server MCP è dotato di un client Amazon Bedrock Identity AgentCore OAuth
7. Il server MCP rende disponibili gli strumenti associati sull'endpoint /mcp affinché gli agenti possano scoprirli.
8. Amazon CloudWatch raccoglie parametri e log operativi dalle implementazioni di server MCP per il monitoraggio e la risoluzione dei problemi.

## Caso d'uso di Agent Builder

Descrive l'architettura di Agent Builder




Il flusso di processo di alto livello per i componenti di Agent Builder distribuiti con il CloudFormation modello AWS è il seguente:

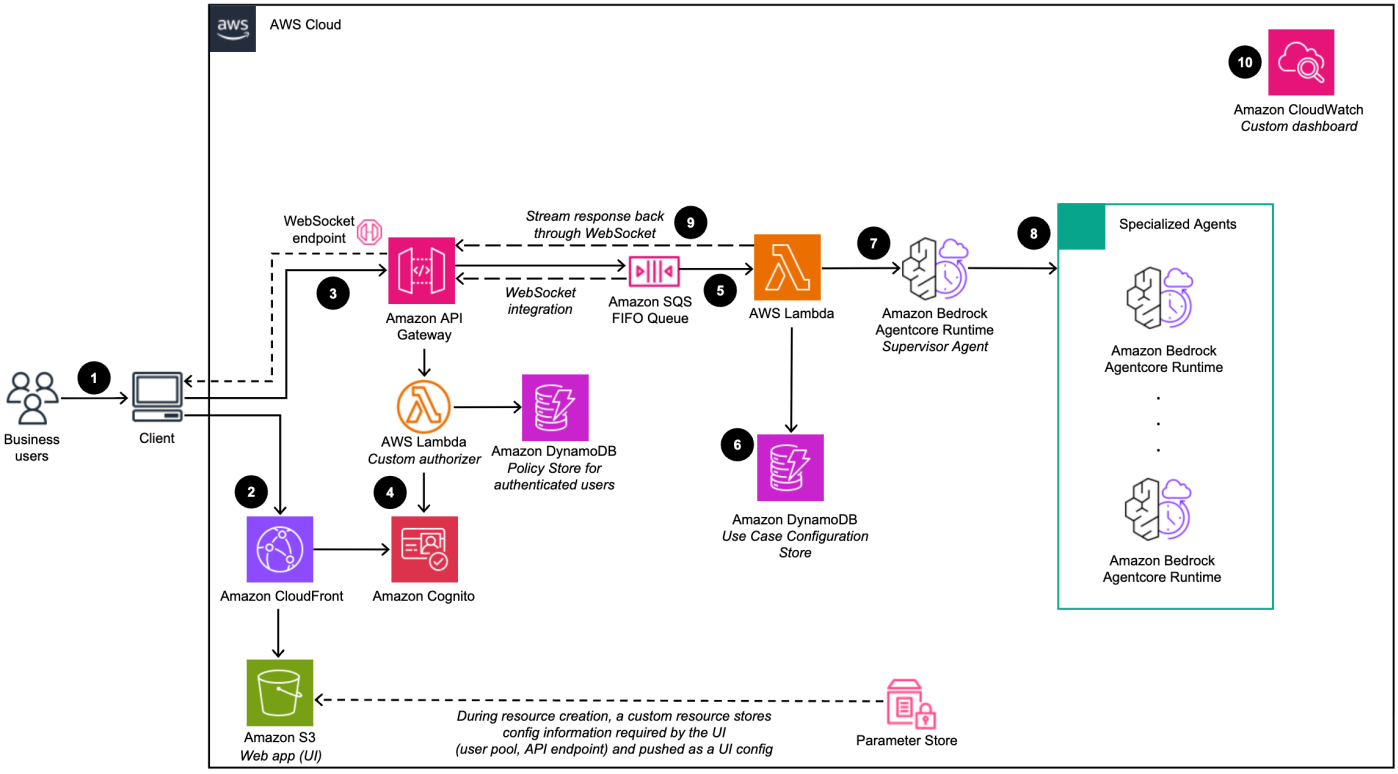
1. Gli utenti amministratori distribuiscono lo use case utilizzando il Deployment Dashboard. Gli utenti aziendali accedono all'interfaccia utente dello use case.
2. CloudFront fornisce l'interfaccia utente Web ospitata in un bucket S3.
3. L'interfaccia utente Web sfrutta un' WebSocket integrazione creata utilizzando API Gateway. L'API Gateway è supportato da una funzione di autorizzazione Lambda personalizzata, che restituisce la policy AWS Identity and Access Management (IAM) appropriata basata sul gruppo Amazon Cognito a cui appartiene l'utente che effettua l'autenticazione. La policy è archiviata in DynamoDB.
4. Amazon Cognito autentica gli utenti e supporta sia l'interfaccia utente CloudFront Web che l'API Gateway.
5. Le richieste in arrivo dall'utente aziendale vengono passate da API Gateway a una coda Amazon SQS e quindi alla funzione AWS Lambda. La coda consente il funzionamento asincrono dell'integrazione tra API Gateway e Lambda. La coda passa le informazioni di connessione alla funzione Lambda che invierà quindi i risultati direttamente alla connessione websocket API Gateway per supportare chiamate di inferenza a lunga durata.
6. La funzione AWS Lambda recupera la configurazione dell'agente da DynamoDB.

7. Utilizzando l'input dell'utente e tutte le configurazioni dei casi d'uso pertinenti, la funzione AWS Lambda crea e invia un payload di richiesta all'agente, in esecuzione su Amazon Bedrock Runtime. AgentCore
8. L'agente si connette ai server MCP associati e registra gli strumenti nell'istanza dell'agente strands. L'agente quindi seleziona ed esegue autonomamente le azioni in base alle descrizioni degli strumenti e ai requisiti delle attività.
9. Quando la risposta ritorna dal AgentCore runtime di Amazon Bedrock, la funzione Lambda trasmette la risposta attraverso l'API WebSocket Gateway per essere utilizzata dall'applicazione client.

> **Note**
>
> - L'elaborazione dell'agente è limitata al timeout di esecuzione Lambda (15 minuti).

## Caso d'uso di Workflow Builder

Descrive l'architettura di Workflow Builder



Il flusso di processo di alto livello per i componenti Workflow Builder distribuiti con il CloudFormation modello AWS è il seguente:

1. Gli utenti amministratori distribuiscono il flusso di lavoro utilizzando il Deployment Dashboard, selezionando gli agenti di Agent Builder da includere come agenti specializzati.
2. CloudFront fornisce l'interfaccia utente Web ospitata in un bucket S3.
3. L'interfaccia utente Web sfrutta un' WebSocket integrazione creata utilizzando API Gateway. L'API Gateway è supportato da una funzione di autorizzazione Lambda personalizzata, che restituisce la policy AWS Identity and Access Management (IAM) appropriata basata sul gruppo Amazon Cognito a cui appartiene l'utente che effettua l'autenticazione. La policy è archiviata in DynamoDB.
4. Amazon Cognito autentica gli utenti e supporta sia l'interfaccia utente CloudFront Web che l'API Gateway.
5. Le richieste in arrivo dall'utente aziendale vengono passate da API Gateway a una coda Amazon SQS e quindi alla funzione AWS Lambda. La coda consente il funzionamento asincrono dell'integrazione tra API Gateway e Lambda.
6. La funzione AWS Lambda recupera la configurazione del flusso di lavoro da DynamoDB, incluso l'elenco degli agenti Agent Builder specializzati.
7. Utilizzando l'input dell'utente e la configurazione del flusso di lavoro, Lambda invia le richieste ad Amazon Bedrock AgentCore Runtime che ospita l'agente supervisore.
8. L'agente supervisore crea istanze locali di tutti gli agenti Agent Builder specializzati all'interno dell'ambiente Runtime. AgentCore Questi agenti specializzati vengono registrati come strumenti utilizzando il pattern Agents as Tools. Il supervisore seleziona e delega quindi autonomamente il lavoro ad agenti specializzati in base alle descrizioni degli agenti e ai requisiti delle attività.
9. L'agente supervisore aggrega i risultati degli agenti specializzati e formula la risposta finale, restituendola alla Lambda per essere ritrasmessa all'applicazione client tramite l'API Gateway Websocket.

> **Note**
>
> - L'elaborazione del workflow è limitata al timeout di esecuzione Lambda (15 minuti).

# Considerazioni sulla progettazione di AWS Well-Architected

Questa soluzione è stata progettata con le migliori pratiche di AWS Well-Architected Framework che aiutano i clienti a progettare e gestire carichi di lavoro affidabili, sicuri, efficienti ed economici nel cloud.

Questa sezione descrive come sono stati applicati i principi di progettazione e le best practice del Well-Architected Framework durante la creazione di questa soluzione.

## Eccellenza operativa

Questa sezione descrive come abbiamo progettato questa soluzione utilizzando i principi e le migliori pratiche del pilastro dell'eccellenza operativa.

- Abbiamo creato la soluzione infrastructure-as-code utilizzando Amazon CloudFormation.
- Le funzioni Lambda inviano metriche personalizzate CloudWatch e un CloudWatch dashboard personalizzato per monitorare lo stato della soluzione.
- I componenti della soluzione sono altamente modularizzati e offrono la flessibilità di scegliere quali componenti implementare.

## Sicurezza

Questa sezione descrive come abbiamo progettato questa soluzione utilizzando i principi e le migliori pratiche del pilastro della sicurezza.

- La dashboard di distribuzione e tutti i casi d'uso sono autenticati e autorizzati con Amazon Cognito.
- Tutte le comunicazioni tra servizi utilizzano i ruoli AWS IAM.
- Tutti i ruoli della soluzione seguono l'accesso con privilegi minimi; ciò significa che vengono concesse solo le autorizzazioni minime richieste.
- Tutti gli storage di dati, inclusi i bucket S3, DynamoDB e Amazon Kendra, dispongono di crittografia inattiva.

## Affidabilità

Questa sezione descrive come abbiamo progettato questa soluzione utilizzando i principi e le migliori pratiche del pilastro dell'affidabilità.

- Architettura basata sul paradigma serverless.
- Abbiamo creato l'architettura per la scalabilità orizzontale su richiesta e il ripristino automatico in caso di guasto dell'infrastruttura sottostante.
- L'architettura include il buffering e la limitazione delle richieste per non sovraccaricare gli endpoint sottostanti.

## Efficienza delle prestazioni

Questa sezione descrive come abbiamo progettato questa soluzione utilizzando i principi e le migliori pratiche del pilastro prestazione-efficienza.

- La soluzione utilizza DynamoDB, un database NoSQL serverless completamente gestito con scalabilità su richiesta.
- La soluzione utilizza Amazon S3 per lo storage di oggetti e per ospitare un sito Web (tramite CloudFront) per fornire scalabilità a basso costo e una durabilità di 11 secondi.

## Ottimizzazione dei costi

Questa sezione descrive come abbiamo progettato questa soluzione utilizzando i principi e le migliori pratiche del pilastro dell'ottimizzazione dei costi.

- Laddove possibile, abbiamo creato la soluzione per utilizzare l'architettura serverless, in modo da pagare solo per ciò che si utilizza.

## Sostenibilità

Questa sezione descrive come abbiamo progettato questa soluzione utilizzando i principi e le migliori pratiche del pilastro della sostenibilità.

- L'architettura modulare e componibile della soluzione offre la flessibilità necessaria per personalizzare le risorse da fornire per singoli casi d'uso.
- L'architettura utilizza elaborazione e storage senza server, che ottimizzano l'utilizzo delle risorse.
- Essendo una soluzione basata sul cloud, questa soluzione sfrutta risorse condivise, reti, alimentazione, raffreddamento e strutture fisiche.

# Dettagli dell'architettura

Questa sezione descrive i componenti e i servizi AWS che compongono questa soluzione e i dettagli dell'architettura su come questi componenti interagiscono.

## Servizi AWS in questa soluzione

| Servizio AWS | Description |
| --- | --- |
| Gateway Amazon API | Core. Questo servizio fornisce il REST APIs per la dashboard di distribuzione e l' WebSocket API per il caso d'uso. |
| AWS CloudFormation | Nucleo. Questa soluzione è distribuita come CloudFormation modello e CloudFormation distribuisce le risorse AWS per la soluzione. |
| Amazon CloudFront | Nucleo. CloudFront serve i contenuti Web ospitati in Amazon S3. |
| Amazon Cognito | Nucleo. Questo servizio gestisce la gestione e l'autenticazione degli utenti per l'API. |
| Amazon DynamoDB | Nucleo. DynamoDB archivia le informazioni sulla distribuzione e i dettagli di configurazione per la dashboard di distribuzione. Memorizza la cronologia chat e le conversazioni IDs nel caso d'uso Text per abilitare la cronologia delle conversazioni e la disambiguazione delle query. |
| AWS Lambda | Nucleo. La soluzione utilizza le funzioni Lambda per:<br><br>* Supporta gli endpoint WebSocket REST e API * Gestisci la logica di base di ogni use case |

| Servizio AWS | Description |
| --- | --- |
| | orchestrator * Implementa risorse personali zzate durante la distribuzione CloudFormation |
| Amazon S3 | Nucleo. Amazon S3 ospita i contenuti Web statici. |
| Amazon CloudWatch | Supporto. Questa soluzione pubblica i log dalle risorse della soluzione nei CloudWatch registri e pubblica le metriche nelle metriche. CloudWatch La soluzione crea anche una dashboard per visualizzare questi datiCloud Watch . |
| AWS Systems Manager | Supporto. Systems Manager fornisce il monitoraggio delle risorse a livello di applicazione e la visualizzazione delle operazioni delle risorse e dei dati sui costi. Utilizzato anche per memorizzare i dati di configurazione in Parameter Store. |
| AWS WAF | Supporto. AWS WAF viene distribuito prima della distribuzione API Gateway per proteggerlo. |
| Amazon Bedrock | Facoltativo. La soluzione sfrutta Amazon Bedrock per accedere a modelli base o personalizzati, Amazon Bedrock Agents, Amazon Bedrock Knowledge Base. Amazon Bedrock è l'integrazione consigliata per impedire che i dati lascino la rete AWS. |
| Amazon Bedrock AgentCore | Opzionale La soluzione sfrutta Amazon Bedrock AgentCore per eseguire e supportare connessioni a server MCP, nonché casi d'uso di Agent Builder e Workflow. |

| Servizio AWS | Description |
| --- | --- |
| Amazon Elastic Container Registry (Amazon ECR) | Facoltativo. Per le implementazioni di Agent Builder, ECR archivia e distribuisce le immagini dei container degli agenti. La soluzione utilizza ECR Pull-Through Cache per recuperare automaticamente le immagini predefinite degli agenti dall'archivio ECR pubblico del team GAAB. |
| AWS Distro per OpenTelemetry (ADOT) | Facoltativo. Per le implementazioni di Agent Builder, ADOT fornisce una strumentazione automatica per l'osservabilità degli agenti, abilitando la tracciabilità distribuita e la registrazione strutturata per le operazioni degli agenti. |
| Amazon Kendra | Facoltativo. Nel caso Text use case, gli utenti amministratori possono facoltativamente decidere di collegare un indice Amazon Kendra da utilizzare come knowledge base per la conversazione con il LLM. Questo può essere usato per inserire nuove informazioni nel LLM, dandogli la possibilità di utilizzare tali informazioni nelle sue risposte. |
| Amazon SageMaker AI | Facoltativo. La soluzione può integrarsi con un endpoint di inferenza Amazon SageMaker AI per accedere ospitato all'interno del tuo account e della tua regione AWS ed è un'integrazione preferita per evitare FMs che i tuoi dati lascino la rete AWS.<br><br>**Note**<br>È necessario distribuire la soluzione nella stessa regione in cui è disponibile l'endpoint di inferenza. |

| Servizio AWS | Description |
| --- | --- |
| Amazon Virtual Private Cloud | Facoltativo. La soluzione offre la possibilità di implementare componenti con una configura zione abilitata per VPC. Durante l'impleme ntazione della soluzione con una configura zione abilitata per VPC, hai la possibilità di lasciare che la soluzione crei un VPC per te o utilizzare un VPC esistente nello stesso account e nella stessa regione in cui verrà distribuita la soluzione (Bring Your Own VPC). Se la soluzione crea il VPC, crea i componenti di rete necessari che includono sottoreti, gruppi di sicurezza e relative regole, tabelle di routing, rete, gateway NAT ACLs, gateway Internet, endpoint VPC e relative politiche. |

# Dashboard di implementazione

## Autorizzatori personalizzati API Gateway

In apparenza, gli autorizzatori personalizzati Lambda per API Gateway vengono utilizzati per tutte le chiamate API ( RESTful entrambe WebSocket e basate) per verificare se un determinato utente è autorizzato a eseguire un'azione in base ai gruppi a cui appartiene. Questo autorizzatore personalizzato è supportato da una tabella DynamoDB contenente le politiche per ogni gruppo. Quando viene richiamata un'API, API Gateway richiama la funzione di autorizzazione personalizzata Lambda, che decodifica il token di accesso Amazon Cognito fornito per determinare a quali gruppi di utenti appartiene l'utente. La tabella delle politiche viene quindi interrogata in base al nome del gruppo per restituire la politica pertinente per quel gruppo.

Ad ogni nuova implementazione di un caso d'uso, la policy di amministrazione viene aggiornata per memorizzare una nuova istruzione che consente l'azione Execute-API:Invoke sull'API di quel caso d'uso. Quando i casi d'uso vengono eliminati, l'istruzione corrispondente viene rimossa dalla policy.

Per i gruppi creati per un singolo caso d'uso, nella policy è presente una sola istruzione, che consente l'azione Execute-API:Invoke solo sull'API di quel caso d'uso.

Grazie a questa struttura, qualsiasi utente appartenente al gruppo di un caso d'uso può accedere all'API di quel caso d'uso. Un singolo utente può anche essere aggiunto manualmente a più gruppi per consentire a quell'utente di utilizzare più casi d'uso.

> **Warning**
>
> Puoi anche modificare manualmente le politiche per un determinato gruppo nella tabella delle politiche se desideri concedere l'accesso a un nuovo caso d'uso a un gruppo di utenti esistente. Il gruppo di casi d'uso viene eliminato quando lo use case viene eliminato (anche se sono state apportate modifiche manuali), quindi procedi con cautela quando elimini un caso d'uso.

Nel caso in cui uno stack di casi d'uso venga distribuito in modo indipendente (senza l'uso della dashboard di distribuzione), viene creato un pool di utenti Amazon Cognito per quella distribuzione contenente un singolo utente con accesso all'API di quel caso d'uso. Questo pool di utenti appartiene solo a questo caso d'uso e non è condiviso tra altre distribuzioni autonome.

# Caso di utilizzo del testo

## Supporto per lo streaming

In un'applicazione di chat, la latenza è una metrica importante per consentire un'esperienza utente reattiva. La possibilità che le inferenze LLM richiedano da secondi a minuti pone sfide su come offrire al meglio i contenuti ai clienti. Per questo motivo, diversi provider LLM consentono lo streaming delle risposte al chiamante. Invece di attendere il completamento dell'intera inferenza prima di restituire una risposta, ogni token può essere restituito quando è disponibile.

Per supportare l'uso di questa funzionalità, il caso d'uso Text è stato progettato per utilizzare un' WebSocket API a supporto dell'esperienza di chat. Questo WebSocket viene distribuito tramite API Gateway. L'uso di un' WebSocket API consente di creare una connessione all'inizio di una sessione di chat e di trasmettere le risposte attraverso quel socket. Ciò consente alle applicazioni frontend di fornire un'esperienza utente migliore.

> **Note**
>
> Anche se un modello fornisce supporto per lo streaming, ciò non significa necessariamente che la soluzione sarà in grado di trasmettere le risposte tramite l' WebSocket API. È

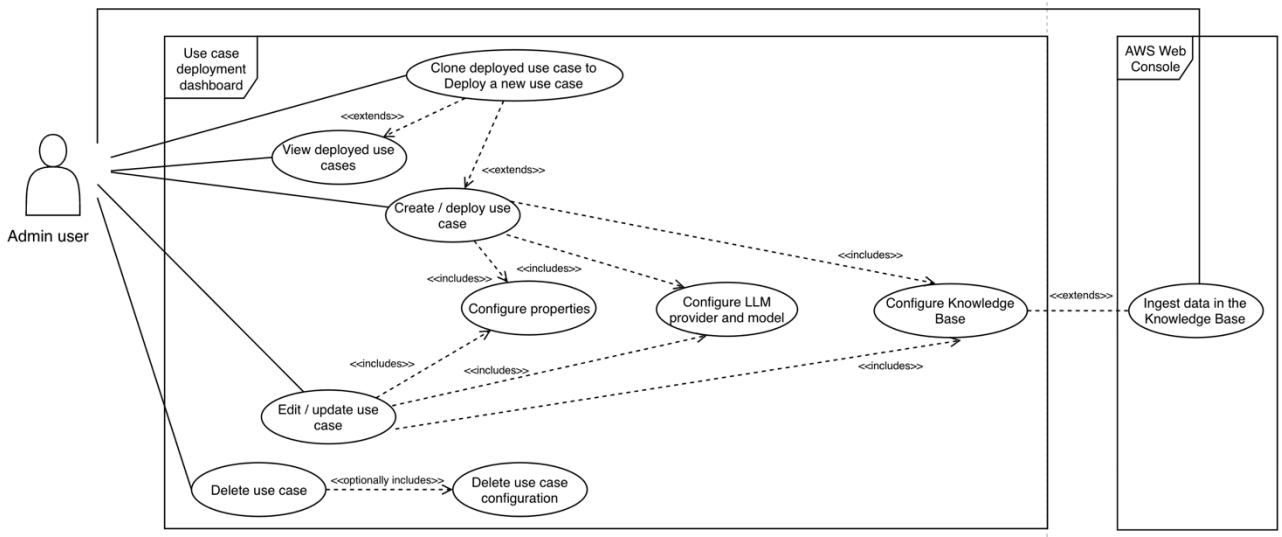
necessario che la soluzione abiliti la logica personalizzata per supportare lo streaming per ogni fornitore di modelli. Se lo streaming è disponibile, gli utenti amministratori potranno utilizzare enable/disable questa funzionalità al momento dell'implementazione.

# Come funziona la soluzione Generative AI Application Builder on AWS

L'utente amministratore si interfaccia principalmente con la dashboard di Deployment per visualizzare, creare e gestire implementazioni di use case nuove ed esistenti. Tramite questa dashboard, l'utente amministratore ha accesso alle seguenti azioni:

- Visualizza l'elenco delle distribuzioni
- Crea nuove distribuzioni
- Modifica le distribuzioni esistenti
- Clona la configurazione di una distribuzione per crearne una nuova
- Eliminare una distribuzione (rimuovere il provisioning delle risorse tramite un' CloudFormation eliminazione)
- Eliminare definitivamente i dettagli di configurazione di una distribuzione

Illustra il diagramma dei casi d'uso per l'utente amministratore del dashboard di distribuzione

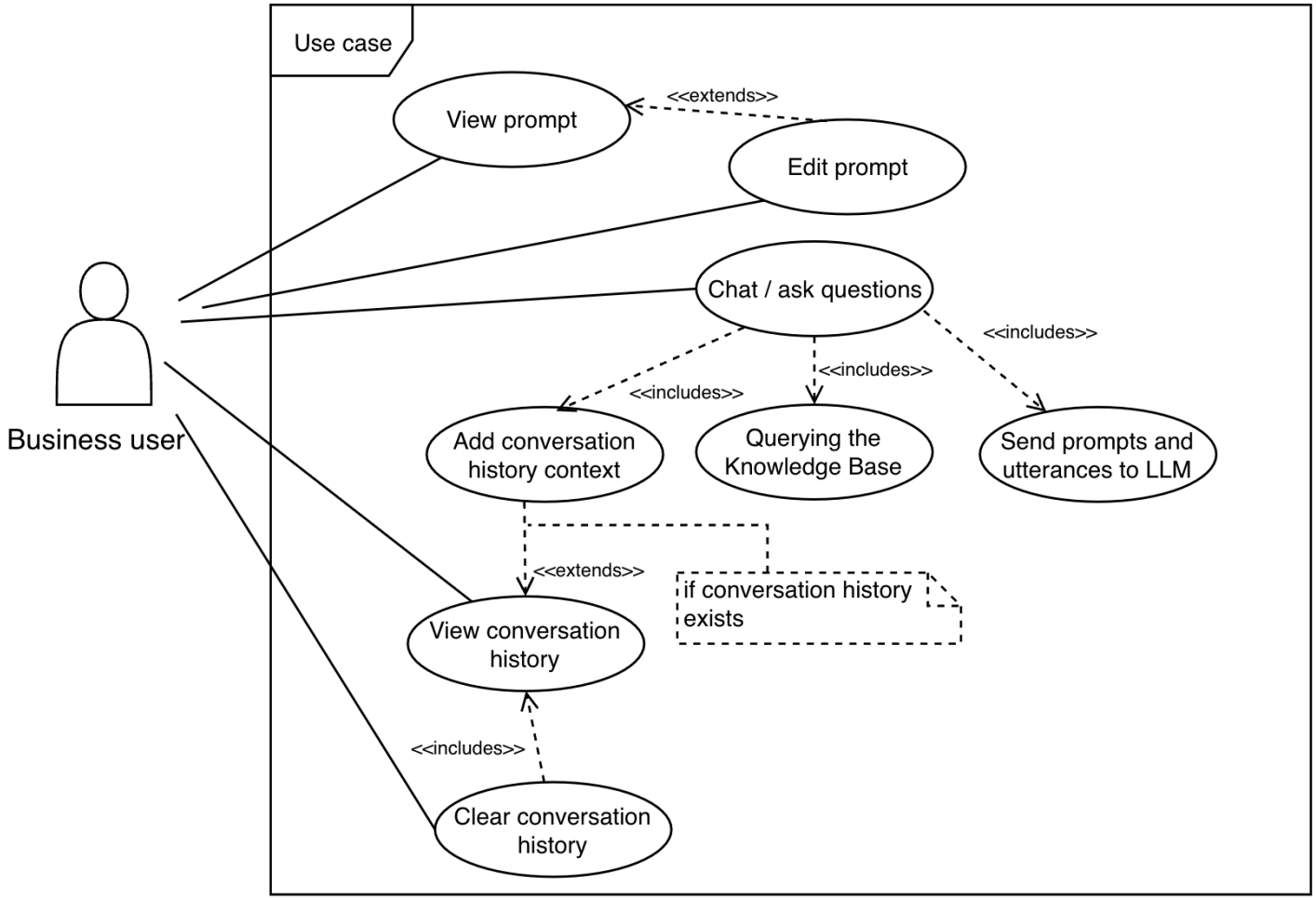


> **Note**
>
> L'utente amministratore potrebbe non avere accesso diretto alla console AWS. In tal caso, l'utente amministratore deve collaborare con l' DevOps utente per supportare azioni come l'inserimento di dati in una knowledge base di Kendra.

Nel caso d'uso Text, l'utente aziendale ha accesso a un'interfaccia utente che gli consente di chattare con l'LLM. Le specifiche di questa configurazione sono controllate dalle impostazioni di distribuzione configurate dall'utente amministratore. Nel caso Text use case, l'utente aziendale ha accesso alle seguenti azioni:

- Invia messaggi tramite l'interfaccia di chat
- Visualizza la cronologia delle conversazioni
- Cancella la cronologia delle conversazioni
- Visualizza prompt
- Richiesta di modifica

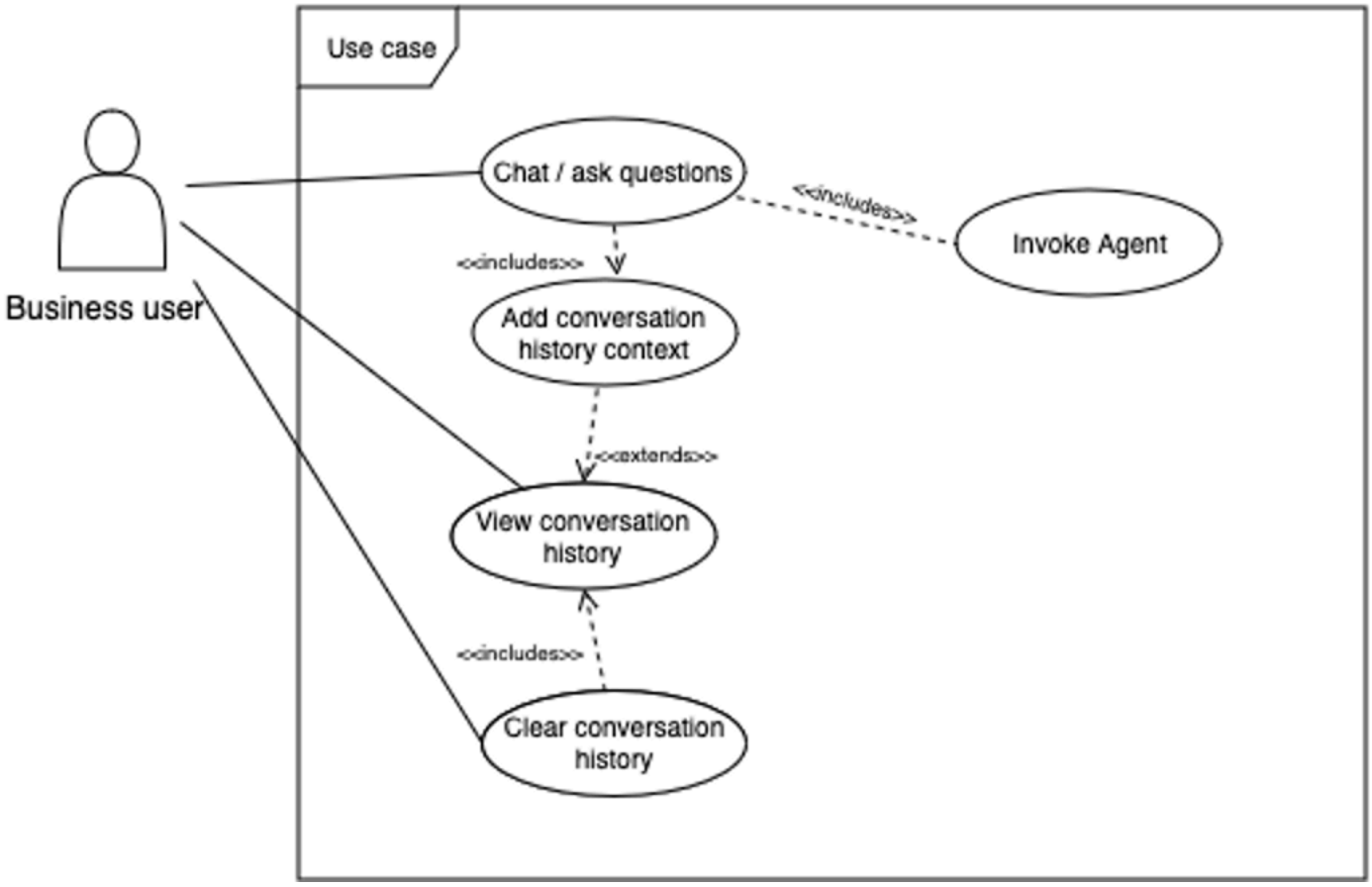
Rappresenta il diagramma dei casi d'uso per l'utente aziendale dello use case Text




Con lo use case Bedrock Agent, l'utente aziendale può accedere a un'interfaccia utente per chattare con l'Amazon Bedrock Agent configurato. L'utente amministratore può configurare queste specifiche nelle impostazioni di distribuzione. Nel caso d'uso di Bedrock Agent, l'utente aziendale ha accesso alle seguenti azioni:

- Invia messaggi tramite l'interfaccia di chat
- Visualizza la cronologia delle conversazioni
- Cancella la cronologia delle conversazioni

Illustra il diagramma dei casi d'uso per l'utente aziendale dello use case di Bedrock Agent




# Agent Builder

Agent Builder fornisce una piattaforma per la creazione, la distribuzione e la gestione di agenti AI pronti per la produzione su Amazon Bedrock. AgentCore Questa sezione descrive i componenti tecnici e i dettagli di implementazione.

## AgentCore integrazione

Agent Builder utilizza un approccio di distribuzione basato sulla configurazione con immagini predefinite degli agenti per consentire implementazioni rapide, sicure e scalabili.

Immagini predefinite degli agenti

Le immagini dei container degli agenti vengono create dal team GAAB durante la CI/CD pipeline e pubblicate in un repository ECR pubblico. Ogni versione dell'immagine è collegata alla versione della

soluzione GAAB (ad esempio, v4.0.0 →:v4.0.0). gaab-strands-agent Le immagini sono basate su Strands SDK e includono:

- Ambiente di runtime dell'agente
- Integrazione del client MCP
- Funzionalità di gestione della memoria
- OpenTelemetry strumentazione

### Cache pull-through ECR

La soluzione utilizza ECR Pull-Through Cache per distribuire automaticamente le immagini degli agenti dall'archivio ECR pubblico all'ECR privato del cliente. Questo servizio gestito da AWS:

- Memorizza le immagini nella cache alla prima estrazione (ritardo di 2-5 minuti)
- Elimina la logica di copia personalizzata delle immagini
- Fornisce la disponibilità locale delle immagini per le distribuzioni successive
- Crea regole di cache uniche per ogni distribuzione per evitare conflitti

### Archiviazione della configurazione

Le configurazioni degli agenti sono archiviate in DynamoDB insieme alle configurazioni dei casi d'uso esistenti. Ogni configurazione include:

- Modello di prompt di sistema
- Fornitore e ID del modello
- Parametri del modello (temperatura, max_tokens)
- Riferimenti ed endpoint del server MCP
- Impostazioni della memoria (commutazione della memoria a lungo termine)
- Metadati di distribuzione

### Registro delle versioni dell'immagine

Una tabella DynamoDB tiene traccia delle versioni delle immagini degli agenti disponibili e della relativa URIs cache, abilitando la gestione delle versioni e la compatibilità con le versioni precedenti.

# Configurazione dell'agente

### Richieste di sistema

I prompt di sistema definiscono il comportamento, la personalità e le capacità degli agenti. Gli utenti amministratori possono:

- Modificare il modello predefinito tramite l'interfaccia utente di Agent Builder
- Include istruzioni per l'utilizzo degli strumenti e la formattazione delle risposte
- Ripristina il modello predefinito in qualsiasi momento

### Selezione del modello

Agent Builder supporta i modelli Amazon Bedrock nella versione 4.0.0:

- Fornitore del modello: Amazon Bedrock (solo opzione nella versione 4.0.0)
- Selezione del modello: Claude, Nova e altri modelli Bedrock
- Parametri del modello: Temperature, max_tokens, top_p e impostazioni specifiche del modello

### Integrazione con server MCP

I server Model Context Protocol forniscono agli agenti l'accesso a strumenti e dati aziendali:

- Individuazione dei server tramite l'endpoint API GET /mcp
- Configurazione dinamica senza modifiche al codice
- Autenticazione e gestione degli endpoint
- Esposizione delle funzionalità degli strumenti agli agenti

# Streaming ed elaborazione

### Streaming in tempo reale

Agent Builder utilizza Server-Sent Events (SSE) da AgentCore bridge a per WebSocket lo streaming di risposte in tempo reale:

- La funzione Lambda stabilisce la connessione SSE a Runtime AgentCore

- Gli stream sono collegati all'API Gateway WebSocket
- Consente la consegna delle token-by-token risposte ai clienti
- Mantiene la connessione per le richieste di lunga durata

### Vincoli di elaborazione

L'elaborazione dell'agente nella v4.0.0 è limitata al timeout di esecuzione Lambda:

- Tempo massimo di elaborazione: 15 minuti
- Modello di elaborazione sincrona
- Adatto per agenti conversazionali e flussi di lavoro moderati
- Supporto asincrono esteso previsto per la versione 4.1+

## Gestione della memoria

### Memoria a breve termine

Abilitato per impostazione predefinita per tutti gli agenti che utilizzano una configurazione personalizzata MemoryHookProvider:

- Cattura gli eventi di conversazione tramite i gestori di callback di Strands
- Organizza per actorID e sessionID per l'isolamento del contesto
- Mantiene il contesto della conversazione all'interno delle sessioni
- Integrazione automatica con la AgentCore memoria

### Memoria a lungo termine

Funzionalità opzionale che utilizza AgentCore Memory Tool di strands_tools:

- Attivazione semplice nell'interfaccia utente di Agent Builder
- Strategia di memoria semantica con impostazioni predefinite
- Accesso controllato dall'agente tramite invocazione naturale degli strumenti
- Memorizza le informazioni estratte tra le sessioni
- Utilizza ConversationID come SessionID

# Osservabilità

OpenTelemetry Distribuzione AWS (ADOT)

Gli agenti vengono dotati automaticamente di strumenti durante la creazione del container:

- Generazione automatica di tracce per le operazioni degli agenti
- Tracciamento distribuito attraverso i confini del servizio
- Registrazione strutturata con correlazione IDs
- Integrazione con Transaction Search CloudWatch

Flusso di autenticazione

Gli utenti si autenticano tramite Amazon Cognito con token JWT convalidati da autorizzatori Lambda personalizzati che recuperano le policy IAM da DynamoDB in base ai gruppi di utenti.

# Workflow Builder

Workflow Builder consente l'orchestrazione multiagente creando un agente supervisore che coordina più agenti Agent Builder utilizzando il modello di delega Agents as Tools.

## Architettura del workflow

Componenti chiave

- Agente supervisore: agente Entrypoint che riceve le richieste degli utenti e delega ad agenti specializzati
- Agenti specializzati: Agent Builder utilizza casi registrati come strumenti per il supervisore
- Agent Registry: tabella DynamoDB che memorizza le configurazioni e i metadati degli agenti
- Livello di orchestrazione: implementazione del pattern Agents as Tools tramite Strands SDK

## Istanziazione dell'agente

Creazione di agenti locali

Tutti gli agenti specializzati vengono istanziati localmente all'interno dello stesso AgentCore Runtime:

1. Recupera le configurazioni degli agenti da DynamoDB

2. Crea istanze locali di ogni agente Agent Builder
3. Ogni agente mantiene le proprie connessioni al server MCP
4. L'agente supervisore registra agenti specializzati come strumenti
5. Strands SDK gestisce la selezione e la delega degli agenti

# Pianifica la tua implementazione

Questa sezione descrive le considerazioni relative a costi, sicurezza, regione e quota per la pianificazione della distribuzione.

> **Important**
>
> Questa soluzione sfrutta Amazon Bedrock come servizio principale per l'accesso ai modelli generati dall'intelligenza artificiale. Devi prima richiedere l'accesso ai modelli prima che siano disponibili per l'uso all'interno della soluzione. Per i dettagli, consulta Model access nella Amazon Bedrock User Guide.

## Regioni AWS supportate

> **Important**
>
> Questa soluzione utilizza facoltativamente i servizi Amazon Bedrock e Amazon Kendra, che attualmente non sono disponibili in tutte le regioni AWS. È necessario avviare questa soluzione in una regione AWS in cui questi servizi sono disponibili. Per la disponibilità più aggiornata dei servizi AWS per regione, consulta l'AWS Regional Services List.

Generative AI Application Builder su AWS è supportato nelle seguenti regioni AWS:

| Nome della Regione | |
|---|---|
| Stati Uniti orientali (Ohio) | Canada (Centrale) |
| Stati Uniti orientali (Virginia settentrionale) | Europa (Francoforte) |
| Stati Uniti occidentali (California settentrionale) | Europa (Irlanda) |
| Stati Uniti occidentali (Oregon) | Europa (Londra) |
| Asia Pacifico (Mumbai) | Europa (Milano) |
| Asia Pacifico (Seul) | Europa (Parigi) |

| Nome della Regione | |
|---|---|
| Asia Pacifico (Singapore) | Europa (Stoccolma) |
| Asia Pacifico (Sydney) | Medio Oriente (Bahrein) |
| Asia Pacifico (Tokyo) | Sud America (San Paolo) |

> **Note**
>
> Se utilizzi un modello base accessibile al di fuori di AWS nelle tue distribuzioni, verifica con il fornitore del modello in quali regioni APIs sono disponibili. Se APIs sono disponibili solo in determinate regioni, potresti riscontrare instabilità sotto forma di latenza elevata o addirittura di timeout. È inoltre importante rivolgersi ai team legali e di conformità dell'organizzazione per valutare le considerazioni relative all'attraversamento dei confini regionali da parte dei dati.

# Costo

Con questa soluzione AWS, paghi solo per le risorse che utilizzi e non sono previste tariffe minime o costi di configurazione. Gli utenti pagano per la dashboard utilizzata per avviare i casi d'uso dell'intelligenza artificiale generativa e per tutti i casi d'uso implementati. Il costo dei casi d'uso implementati dipende dalle configurazioni. Configurazioni di esempio:

1. Una semplice dashboard di implementazione che costa circa 20 USD al mese.
2. Un semplice caso d'uso di chatbot pronto per la produzione distribuito con impostazioni predefinite in esecuzione negli Stati Uniti orientali (Virginia settentrionale), alimentato da Amazon Bedrock senza accesso ai documenti, che costa anche circa 200 dollari al mese.
3. Un sistema scalabile in un caso d'uso di Amazon VPC che supporta 8.000 query al giorno su decine di migliaia di documenti, con un costo di circa 1.500 USD al mese. Il costo dello use case varierà a seconda della configurazione, ad esempio casi d'uso in formato Text con diversi fornitori di modelli, con o senza Retrieval Augmented Generation (RAG) abilitato e così via.

| Descrizione del carico di lavoro | Costo stimato (USD/mese) |
| --- | --- |
| Esempio di costo per la dashboard di implementazione | 20 USD al mese |
| Esempi di costi per un proof of concept basato su testo<br><br>(include dashboard di implementazione e 1 caso d'uso testuale, circa 100 interazioni al giorno) | 40 USD/mese |
| Esempio dei costi per un motore di query AI generativo altamente scalabile<br><br>(Include dashboard di implementazione, 1 case d'uso testuale e un Amazon Kendra Index for RAG), fino a 100.000 documenti con circa 8.000 query al giorno, con VPC abilitato | 1.500 USD al mese |
| Esempio di costi per un proof of concept basato su agenti<br><br>(Include dashboard di implementazione, 1 caso d'uso di 1 agente Bedrock con Amazon Bedrock Knowledge Bases e Amazon Bedrock Guardrails abilitati, circa 100 interazioni al giorno) | 840 USD/mese |
| Esempi di costi per MCP Server<br><br>(Include dashboard di implementazione, 1 caso d'uso del server MCP con metodo Gateway per l'integrazione Lambda, circa 100 chiamate di strumenti al giorno) | 22 USD al mese |
| Esempi di costi per Agent Builder | $55 al mese |

| Descrizione del carico di lavoro | Costo stimato (USD/mese) |
| --- | --- |
| (Include dashboard di distribuzione, 1 caso d'uso di Agent Builder con integrazione MCP e memoria a lungo termine abilitata, circa 100 interazioni al giorno) | |
| Esempi di costi per Workflow Builder<br><br>(Include dashboard di implementazione, 1 flusso di lavoro con 3 agenti Agent Builder, circa 100 interazioni al giorno) | 109 USD/mese |

**Important**

Questi esempi hanno il solo scopo di aiutarti a stimare i costi per i tuoi carichi di lavoro specifici. L'uso di diverse LLMs configurazioni o servizi AWS può modificare i costi (ad esempio, serverless/on-demand billing vs. provisioned/time -billed). Per gestire i costi, consigliamo di creare un budget tramite AWS Cost Explorer. I prezzi sono soggetti a modifiche. Per tutti i dettagli, consulta la pagina web dei prezzi per ogni servizio AWS utilizzato in questa soluzione.

## Esempi di costi per l'esecuzione della dashboard di distribuzione

La tabella seguente fornisce la ripartizione dei costi per un dashboard di distribuzione con parametri predefiniti e 100 utenti attivi nella regione Stati Uniti orientali (Virginia settentrionale) per un mese, il che costerà circa 20 USD al mese.

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Gateway API, DynamoDB, CloudFront Amazon S3, Lambda, archivio parametri di Systems Manager | 5.000 chiamate API REST da 512 KB al mese senza caching abilitato | $1,97 |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Amazon Cognito | 100 utenti attivi al mese con funzionalità di sicurezza avanzate abilitate e nessun utente che accede tramite la federazione SAML o OIDC | $5,55 |
| AWS WAF | 10.000 richieste web su 1 ACL web e 7 regole definite senza gruppi di regole | $12,60 |
| Costo totale del dashboard di implementazione | | 20,12 USD |

## Esempi di costi per un proof of concept basato su testo

Un dashboard di distribuzione può avere molti casi d'uso implementati contemporaneamente. La tabella seguente mostra la ripartizione dei costi di un caso d'uso implementato senza RAG per 1 utente aziendale che esegue 100 query al giorno con il LLM. Le query vengono inviate come messaggio di testo su WebSocket e la risposta viene trasmessa in streaming come token, presupponendo che lo streaming sia abilitato. Utilizzando il modello Amazon Bedrock Nova Pro, il costo di esecuzione di questo use case è di circa 20 USD al mese.

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, AWS Systems Manager Parameter Store | 100 interazioni in chat al giorno. Dimensione media dei messaggi 32 KB per messaggio e 5 minuti per connessione. | $0,61 |
| CloudWatch | CloudWatch Registri da 1,5 GB con modalità dettagliata attiva per la sperimentazione | 7,23$ |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Amazon DynamoDB | Tabella della cronologia delle conversazioni, 1 GB di spazio di archiviazione<br><br>Tabella di configurazione LLM, 1 GB di spazio di archiviazione | 3,05$ |
| Subtotale dei costi del caso d'uso (esclusi) LLMs | | $10,89 |
| Amazon Bedrock (Nova Pro) | Ipotesi per 100 interazioni al giorno:<br><br>* Costo mensile per 190.000 token di input al giorno = 0,152 USD × 30 * Costo mensile per 16.000 token di output al giorno = 0,0512 × 30 USD | $6,10 |
| Costo totale dell'applicazione con Amazon Bedrock (Nova Pro) | $10,89 (costo del caso d'uso) + $6,10 (costo Amazon Bedrock) | $17,00 |

> **Note**
>
> I costi delle chiamate di inferenza effettuate a servizi esterni alla rete AWS non sono inclusi in queste stime. Consulta la guida ai prezzi del tuo provider LLM se non utilizzi un provider di modelli AWS.
> Le guide ai prezzi per i servizi AWS sono disponibili all'indirizzo: prezzi Amazon Bedrock e prezzi Amazon SageMaker AI.

# Esempio dei costi per un motore di query AI generativo altamente scalabile

La tabella seguente fornisce la ripartizione dei costi di un caso d'uso compatibile con RAG con il modello Nova Pro di Amazon Bedrock come LLM. Quando viene aggiunta una Bedrock Knowledge Base, questo caso d'uso costa circa 1300 dollari al mese

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Gateway API (WebSocket) | 8000 interazioni via chat al giorno. Dimensione media dei messaggi 32 KB per messaggio e 5 minuti per connessione. | 38,89 USD |
| CloudFront | 240.000 richieste al mese con 100 GB di dati trasferiti su Internet e 1 GB di dati trasferit i all'origine | 8,76$ |
| Amazon Bedrock (Nova Pro) | Ipotesi:<br><br>Token di input = PromptTem plate (400) + context (400) + ChatHistory (1080) + query Token di input (20) = 1.900<br><br>Token di output = 160 (media)<br><br>Con 8.000 transazioni al giorno,<br><br>Costo giornaliero dei token di input (1.900 x 8.000 = 15.200.000 token x 0,0008/10 00 prezzo per token)<br><br>Costo giornaliero dei token di output (160 x 8.000 = | 487,80$ |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| | 1.280.000 token x 0,0032/1000 prezzo per token)<br><br>Costo mensile (($12,16 + $4,10) x 30) | |
| CloudWatch | 24 metriche che utilizzano 5 GB di dati inseriti per i log e 1 dashboard | 9,72$ |
| DynamoDB | Tabella DynamoDB per tenere traccia della cronologia delle conversazioni con ogni record fino a 1 KB di dati, 8.000 letture e scritture al giorno | $11,70 |
| Lambda | Dimensioni del contenitore: 128 MB, 512 MB temporanei<br><br>storage, 2 funzioni Lambda utilizzate per l'autorizzazione<br><br>Dimensioni del contenitore: 256 MB, 512 MB di storage temporaneo, 5 richieste al secondo con un tempo di elaborazione medio di 20 secondi | 20,89 USD |
| Costo totale del caso d'uso | | 577,76 USD/mese + costo della Knowledge Base (vedi sotto) |

> **Note**
>
> I costi delle chiamate API effettuate a qualsiasi servizio esterno alla rete AWS non sono inclusi in queste stime. Consulta la guida ai prezzi del tuo provider LLM se non utilizzi Amazon Bedrock.

## Costi per l'aggiunta di una knowledge base

I costi della Knowledge Base varieranno in base al tipo di knowledge base utilizzata e (nel caso di Bedrock) all'archivio vettoriale di supporto utilizzato dalla knowledge base. Il provisioning e la gestione delle knowledge base non rientrano nell'ambito della soluzione.

Basi di conoscenza di Amazon Bedrock

La soluzione non gestisce o fornisce risorse relative alle Knowledge Base di Amazon Bedrock. Amazon Bedrock non prevede costi per l'utilizzo della funzionalità della knowledge base stessa, tuttavia ti verrà addebitato l'utilizzo del modello di incorporamento utilizzato dal tuo caso d'uso per ogni query. Inoltre, l'archivio vettoriale di supporto per la tua knowledge base (ad esempio, un indice in Amazon OpenSearch Service o un database all'interno di Amazon Relational Database Service) avrà un costo associato che non può essere fornito o calcolato qui.

Per lo scenario del motore di query AI generativo ad alta scalabilità di cui sopra, i costi sostenuti da questo servizio per richiamare il modello di incorporamento di Amazon Bedrock sono i seguenti:

| Servizio AWS | Dimensioni | Costo [USD] |
|---|---|---|
| Amazon Bedrock (Amazon Titan Text Embeddings V2) | 8.000 query al giorno con 1.900 token di input per query = 15.200.000 token = 0,30 USD al giorno.<br><br>Costo giornaliero x 30 giorni = costo mensile di 9,00 USD | 9,00 USD |
| Esempio OpenSearch di utilizzo di Amazon Service (Serverless) | Configurazione serverles s di base con 4 unità di OpenSearch calcolo (OCU) | 691,20$ |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| | (minimo fatturabile) = 23,04 USD al giorno<br><br>Costo giornaliero x 30 giorni = 691,20 USD<br><br>**Note**<br>Si tratta di una stima approssimativa, in quanto alcuni carichi di lavoro richiederanno di più OCUs, mentre i clienti con OpenSearch risorse già disponibili dovranno sostenere costi inferiori in questo caso. | |
| Costo aggiuntivo totale | | 700,20$ |

Amazon Kendra

La soluzione può fornirti un indice Kendra oppure puoi portarne uno tuo. Il costo per l'esecuzione di una configurazione adatta al suddetto motore di query AI generativo altamente scalabile di cui sopra è il seguente:

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Amazon Kendra | 0-8.000 richieste al giorno e fino a 100.000 documenti con Amazon Kendra Enterprise Edition con 0-50 fonti di dati | 1.008,00 USD |

> **Note**
>
> Puoi condividere l'indice Amazon Kendra tra diversi casi d'uso, ma ciò può aumentare il numero di query per indice. Se non rientra nell'edizione Amazon Kendra Enterprise, verranno applicati costi aggiuntivi.

## Costo incrementale dell'abilitazione di Amazon VPC per un caso d'uso

La tabella seguente fornisce la ripartizione dei costi di abilitazione di Amazon VPC per uno use case distribuito in due. AZs

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Gateway Amazon NAT | Presupposto: implement azione 2 AZ, con un gateway NAT in ogni AZ. 100 GB di dati elaborati tramite NAT Gateway 730 ore, 100 GB di dati elaborati al mese | 74,70$ |
| AWS PrivateLink (endpoint VPC) | Ipotesi: implementazione 2 AZ, con 1 sottorete privata in ogni AZ e 1 endpoint VPC con 2 interfacce di rete elastiche (). ENIs<br><br>6 endpoint VPC, 2 per endpoint ENIs VPC, 730 ore con 1.024 GB di dati elaborati in un mese | 97,84 USD |
| Indirizzo pubblico IPv4 | Presupposto: 2 implement azioni AZ, 1 sottorete pubblica in ogni AZ con un gateway NAT in ogni sottorete pubblica. Ogni gateway NAT è configura | 7,30 USD |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| | to con 1 gateway pubblico attivo. IPv4<br><br>2 IPv4 indirizzi pubblici attivi x 730 ore al mese x tariffa oraria di 0,005 USD = 7,3 USD | |
| Costo aggiuntivo<br><br>(per Amazon VPC) | | 179,93$ |

## Implicazioni sui costi dell'utilizzo di Provisioned Throughput

I costi di throughput assegnati variano in base al tipo di modello predisposto e al periodo di impegno, nonché alle unità del modello selezionate per il periodo di impegno. L'utilizzo del Provisioned Throughput comporta un costo aggiuntivo.

Per ulteriori informazioni e maggiori informazioni up-to-date sui prezzi, puoi fare riferimento a Bedrock Pricing.

## Costo dell'utilizzo dell'inferenza interregionale

Non sono previsti costi aggiuntivi per il routing o il trasferimento dei dati per l'utilizzo dell'inferenza tra regioni. Paghi lo stesso prezzo per token per i modelli della tua regione di origine o principale.

## Esempio dei costi per un proof of concept basato su agenti

Quando usi Amazon Bedrock Agents, i costi vengono calcolati in base ai componenti che compongono l'agente, come il modello di supporto e la knowledge base (se RAG è abilitato), oltre alle funzionalità aggiuntive che aggiungi. La tabella seguente mostra la ripartizione dei costi di uno use case di Bedrock Agent configurato con un modello Claude 3.5 Sonnet on-demand, Amazon Bedrock Knowledge Bases e Amazon Bedrock Guardrails.

Analogamente al costo per l'aggiunta di Amazon Bedrock Knowledge Bases, questa soluzione non gestisce o fornisce risorse relative agli Amazon Bedrock Agents. Inoltre, la soluzione non comporta costi per l'utilizzo di Amazon Bedrock Knowledge Bases, ma comporta costi per:

- Utilizzo del modello di incorporamento per ogni query che gli viene inviata
- L'archivio vettoriale di supporto per la tua knowledge base (ad esempio, un indice in Amazon OpenSearch Service o un database all'interno di Amazon RDS)

La tabella seguente presuppone 100 interazioni al giorno con 1.900 token di input e 160 token di output per query.

> **Note**
>
> Per questo esempio di utilizzo di Bedrock Agent, se esistesse un gruppo di azioni configurato per utilizzare un'API esterna, tali costi sarebbero aggiuntivi. Non rientrano nell'ambito dei calcoli riportati in questa tabella.

| Servizio AWS | Dimensioni | Costo [USD] |
|---|---|---|
| API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, Systems Manager Parameter Store | 100 interazioni in chat al giorno, dimensione media dei messaggi 32 KB per messaggio, 5 minuti per connessione | $0,61 |
| CloudWatch | CloudWatch Log da 1,5 GB con modalità dettagliata attiva per la sperimentazione | 7,23$ |
| DynamoDB | Tabella di configurazione LLM per record di 1 KB e 1 GB di spazio di archiviazione | $0,25 |
| Subtotale dei costi (esclusi) LLMs | | 8,09 USD |
| Sonetto antropico di Claude 3.5 | * Costo giornaliero per 190.000 token di input al giorno (0,003/1.000 token) = 0,57 USD + | $24,30 |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| | Costo giornaliero × 30 giorni = 17,10 USD* Costo giornaliero per 16.000 token di output al giorno (0,015/1.000 token) = 0,24 USD +<br><br>Costo giornaliero × 30 giorni = 7,20 USD | |
| Amazon Bedrock (Amazon Titan Text Embeddings V2) per le basi di conoscenza Amazon Bedrock | Costo giornaliero per 190.000 token di input al giorno (0,00002/1000 token) = 0,004<br><br>Costo giornaliero × 30 giorni = 0,12 USD | 0,12 USD |
| Esempio OpenSearch di utilizzo di Amazon Service (Serverless) | Configurazione serverless di base con 4 × OpenSearch Compute Unit (OCU) (minimo fatturabile) = 23,04 USD al giorno<br><br>Costo giornaliero × 30 giorni = 691,20 USD | 691,20$ |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Amazon Bedrock Guardrails | I token 190K equivalgono all'incirca a 760.000 (190.000 × 4) caratteri e 3.800 unità di testo (760K caratteri/200)<br><br>Prendi in considerazione un guardrail configurato con filtri di contenuto, filtro per informazioni di identificazione personale (PII), filtro per informazioni sensibili (espressioni regolari) e filtri per parole<br><br>Costo giornaliero del filtro dei contenuti (0,75/1000 unità di testo) + costo del filtro PII (0,1/1000 unità di testo) + filtro per le informazioni sensibili (regex) + filtri di parole = 2,85 $+ 0,38 $0 + $0<br><br>Costo mensile = Costo giornaliero × 30 giorni = 96,90 USD | 96,90$ |
| Costo totale dell'applicazione per un agente supportato da Anthropic Claude 3.5 Sonnet | 8,09 USD (costo del caso d'uso) + 812,52 USD (altre configurazioni di agenti) | 820,61$ |

> **Note**
>
> Consulta la guida ai prezzi del tuo provider LLM se non utilizzi un provider di modelli AWS. Le guide ai prezzi per i servizi AWS sono disponibili all'indirizzo: prezzi Amazon Bedrock e prezzi Amazon SageMaker AI.

# Esempi di costi per MCP Server

I casi d'uso di MCP Server consentono l'implementazione e la gestione di server Model Context Protocol su Amazon AgentCore Bedrock. La tabella seguente mostra la ripartizione dei costi di un caso d'uso del server MCP che utilizza il metodo Gateway per racchiudere le funzioni Lambda esistenti.

La soluzione gestisce l'implementazione e la configurazione del AgentCore Gateway. Ti vengono addebitati i seguenti costi:

- Costi dell'infrastruttura (API Gateway, Lambda, DynamoDB, S3) CloudWatch
- AgentCore Consumo del gateway (per chiamata dello strumento)
- Costi di esecuzione della funzione Lambda (per il metodo Gateway con obiettivi Lambda)
- Costi delle API esterne (per il metodo Gateway con obiettivi API o MCP Server, se applicabile)

| Elemento | Calcoli | Costo |
|---|---|---|
| Amazon API Gateway (API REST) | 100 chiamate di strumenti al giorno × 30 giorni = 3.000 richieste al mese | 0,05 $ |
| AWS Lambda (orchestrazione) | 100 chiamate al giorno × 30 giorni × 1 secondo in media × 512 MB = 3.000 GB di secondo al mese | 0,05 $ |
| Amazon DynamoDB | 3.000 read/write richieste al mese + 1 GB di spazio di archiviazione | 0,15$ |
| Amazon CloudWatch | Monitoraggio e registrazione standard per 3.000 chiamate | $1,00 |
| Simple Storage Service (Amazon S3) | Archiviazione e registri di configurazione (utilizzo minimo) | 0,25 USD |

| Elemento | Calcoli | Costo |
| --- | --- | --- |
| Amazon Bedrock AgentCore Gateway | 3.000 chiamate di strumenti al mese | 0,05 $ |
| Funzione Target Lambda | 100 chiamate al giorno × 30 giorni × 0,5 secondi × 128 MB = 1.500 GB/secondi al mese | 0,25 USD |
| Costo mensile totale | 1,75 USD (infrastruttura) + 0,05 USD (Gateway) AgentCore | 1,80$ |

> **Note**
>
> I costi variano in base al metodo di implementazione (Gateway vs Runtime), ai tipi di destinazione e ai modelli di utilizzo. Le implementazioni del metodo Runtime prevedono costi di AgentCore runtime anziché costi Gateway. I costi delle API esterne e i costi di hosting di container personalizzati sono aggiuntivi.

## Esempi di costi per Agent Builder

Agent Builder ti consente di creare e distribuire agenti personalizzati su Amazon Bedrock. AgentCore La tabella seguente mostra la ripartizione dei costi di un caso d'uso di Agent Builder configurato con Claude 3.5 Sonnet, integrazione del server MCP e memoria a lungo termine abilitata.

La soluzione gestisce la distribuzione e la configurazione del AgentCore Runtime. Ti vengono addebitati i seguenti costi:

- Costi dell'infrastruttura (API Gateway, Lambda, DynamoDB, S3) CloudWatch
- AgentCore Consumo di runtime (ore di CPU e memoria basate sul tempo effettivo di esecuzione dell'agente)
- Inferenza del modello di base (token di input e output)
- AgentCore Memoria (eventi a breve termine e archiviazione/recupero a lungo termine)

La tabella seguente presuppone 100 interazioni al giorno con 1.900 token di input e 160 token di output per query, con un tempo medio di esecuzione dell'agente di 5 secondi per interazione.

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, Systems Manager Parameter Store | 100 interazioni in chat al giorno, dimensione media dei messaggi 32 KB per messaggio, 5 minuti per connessione | $0,61 |
| CloudWatch | CloudWatch Log da 1,5 GB con modalità dettagliata attiva per la sperimentazione | 7,23$ |
| DynamoDB | Tabella di configurazione LLM per record di 1 KB e 1 GB di spazio di archiviazione | $0,25 |
| Subtotale dei costi dell'infr astruttura | | 8,09 USD |
| Runtime di Amazon Bedrock AgentCore | * CPU: 1 vCPU × 5 secondi × 100 interazioni = 125 vCPU-seconds/day = 0.140 vCPU-hours/day + Costo giornaliero: 0,140 × 0,0895$ = 0,013 USD + Costo mensile: 0,013$ × 30 = 0,38 USD<br><br>* Memoria: 512 MB (0,5 GB) × 5 secondi × 100 interazio ni = 250 GB- seconds/day = 0.069 GB-hours/day + Costo giornaliero: 0,069 × 0,00945$ = 0,0007$ + Costo mensile: 0,0007$ × 30 = 0,02 USD | 0,40 USD |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Sonetto antropico di Claude 3.5 | * Costo giornaliero per 190.000 token di input al giorno (0,003/1.000 token) = 0,57 USD+costo giornaliero × 30 giorni = 17,10 USD<br><br>* Costo giornaliero per 16.000 token di output al giorno (0,015/1.000 token) = 0,24 USD + costo giornaliero × 30 giorni = 7,20 USD | $24,30 |
| Memoria Amazon Bedrock AgentCore | * Memoria a breve termine: 100 nuovi eventi events/da y × 0,25 USD/1.000 eventi = 0,025 USD al giorno + Costo mensile: 0,025 USD × 30 = 0,75 USD<br><br>* Archiviazione di memoria a lungo termine (strategi a integrata): 100 record × 0,75/1.000 USD = 0,075 USD al mese records/month<br><br>* Recupero della memoria a lungo termine: 100 retrieval s/day × 0,50 USD/1.000 recuperi = 0,05 USD al giorno + Costo mensile: 0,05 USD × 30 = 1,50 USD | 2,33 USD |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Costo totale dell'applicazione per Agent Builder con Claude 3.5 Sonnet | 8,09 USD (infrastruttura) + 0,40 USD (AgentCor e Runtime) + 24,30 USD (modello) + 2,33 USD (memoria) | $35,12 |

**Note**

AgentCore I prezzi di Runtime si basano sul consumo. I costi effettivi dipendono da:

- Tempo di esecuzione dell'agente (utilizzo della CPU e della memoria durante l'elaborazione attiva)
- Numero di interazioni e relativa complessità
- Utilizzo dello strumento MCP (aggiuntivo CPU/memory per l'esecuzione dello strumento)
- Configurazione della memoria (memoria a breve termine o a lungo termine abilitata)

Per informazioni dettagliate AgentCore sui prezzi, consulta i prezzi di Amazon Bedrock.

**Note**

Se si utilizzano server MCP che richiamano servizi APIs o servizi esterni, tali costi sono aggiuntivi e non rientrano nell'ambito di questo calcolo. Analogamente, se si utilizzano gli strumenti AgentCore Browser o Code Interpreter, si applicano tariffe basate sul consumo pari a 0,0895 USD per vCPU all'ora e 0,00945 USD per GB all'ora.

## Esempi di costi per Workflow Builder

Workflow Builder crea un agente supervisore che orchestra più agenti Agent Builder. La tabella seguente mostra la ripartizione dei costi per un flusso di lavoro con 1 agente supervisore e 3 agenti Agent Builder specializzati, tutti configurati con Claude 3.5 Sonnet e memoria a lungo termine abilitata.

Ipotesi: 100 interazioni al giorno, media di 2 deleghe di agenti per interazione, tempo di esecuzione di 5 secondi per agente.

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, Systems Manager Parameter Store | 100 interazioni in chat al giorno, dimensione media dei messaggi 32 KB per messaggio, 5 minuti per connessione | $0,61 |
| CloudWatch | CloudWatch Log da 1,5 GB con modalità dettagliata attiva per la sperimentazione | 7,23$ |
| DynamoDB | Tabella di configurazione LLM per record di 1 KB e 1 GB di spazio di archiviazione | $0,25 |
| Subtotale dei costi dell'infr astruttura | | 8,09 USD |
| Amazon Bedrock AgentCore Runtime (agente supervisore) | * CPU: 1 vCPU × 5 secondi × 100 interazioni = 0,140 vCPU hours/day × 30 = $0.38 * Memory: 0.5 GB × 5 seconds × 100 interactions = 0.069 GB-hours/day - × 30 = 0,02 USD | 0,40 USD |
| Amazon Bedrock AgentCore Runtime (3 agenti specializ zati) | * Media di 2 delegazioni per interazione = 200 agenti 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 USD | 0,79 USD |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Anthropic Claude 3.5 Sonnet (agente supervisore) | * Ingresso: 190 000$ × 0,003/1.000 $ = 0,57 USD/ giorno tokens/day × 30 = 17,10$ * Uscita: 16 K × 0,015/1.000 $ = 0,24 USD/ giorno × 30 = 7,20$ tokens/da y | 24,30 USD |
| Anthropic Claude 3.5 Sonnet (agenti specializzati) | * Media di 2 delegazioni per interazione * Ingresso: 380 000 × 0,003/1.000 USD = 1,14 USD/giorno tokens/day × 30 = 34,20 USD * Uscita: 32 K × 0,015/1 K = 0,48 USD/giorno × 30 = 14,40 USD tokens/day | 48,60 USD |
| Amazon Bedrock AgentCore Memory (agente supervisore) | * A breve termine: 100 events/ day × 0,25 USD/1.000 × 30 = 0,75 USD * Archiviaz ione a lungo termine: 100 record × 0,75 USD/1.000 USD = 0,08 USD* Recupero a lungo termine: 100 × 0,50 USD/1.000 × 30 = 1,50 USD retrievals/day | 2,33 USD |
| Amazon Bedrock AgentCore Memory (agenti specializzati) | * A breve termine: 200 events/ day × 0,25 USD/1.000 × 30 = 1,50 USD * Archiviaz ione a lungo termine: 200 record × 0,75 USD/1.000 USD = 0,15 USD* Recupero a lungo termine: 200 × 0,50 USD/1.000 × 30 = 3,00 USD retrievals/day | 4,65 USD |

| Servizio AWS | Dimensioni | Costo [USD] |
| --- | --- | --- |
| Costo totale dell'applicazione per Workflow Builder con 3 agenti | 8,09 USD (infrastruttura) + 1,19 USD (AgentCor e Runtime) + 72,90 USD (modelli) + 6,98 USD (memoria) | 89,16$ |

> **Note**
>
> - Tassi di delega più elevati aumentano proporzionalmente il consumo di token
>
> Per informazioni dettagliate AgentCore sui prezzi, consulta i prezzi di Amazon Bedrock.

# Sicurezza

Quando crei sistemi sull'infrastruttura AWS, le responsabilità di sicurezza vengono condivise tra te e AWS. Questo modello di responsabilità condivisa riduce il carico operativo perché AWS gestisce, gestisce e controlla i componenti, tra cui il sistema operativo host, il livello di virtualizzazione e la sicurezza fisica delle strutture in cui operano i servizi. Per ulteriori informazioni sulla sicurezza di AWS, visita AWS Cloud Security.

## Utilizzo di modelli di base su Amazon Bedrock

Amazon Bedrock ospita una raccolta di modelli, dai modelli Amazon Nova agli altri principali modelli di base (FMs). Quando si utilizza Amazon Bedrock, tutti i modelli sono ospitati all'interno dell'infrastruttura AWS. Ciò significa che quando utilizzi Amazon Bedrock come provider LLM, tutte le tue richieste di inferenza rimarranno all'interno della rete AWS e il traffico di rete non lascerà la tua regione.

> **Note**
>
> Tutti i modelli di base (FMs) disponibili tramite Amazon Bedrock sono ospitati direttamente sull'infrastruttura AWS gestita e di proprietà di AWS. I fornitori di modelli non hanno accesso ai dati dei clienti come istruzioni e continuazioni o ai log di servizio di Amazon Bedrock.

Per ulteriori informazioni sul livello di sicurezza di Amazon Bedrock, consulta la sezione Protezione dei dati in Amazon Bedrock nella Guida per l'utente di Amazon Bedrock.

## Ruoli IAM

I ruoli IAM consentono ai clienti di assegnare policy di accesso e autorizzazioni granulari a servizi e utenti sul cloud AWS. Questa soluzione crea ruoli IAM che garantiscono l'accesso alle funzioni Lambda della soluzione per creare risorse regionali.

## CloudWatch Registri

È possibile abilitare la modalità verbosa durante la distribuzione di un caso d'uso utilizzando la pagina di selezione del modello Deployment Dashboard, in Impostazioni aggiuntive. La modalità verbosa consente di registrare CloudWatch registri dettagliati che possono essere utili per il debug e la sperimentazione immediata.

**Note**

Quando la modalità verbosa è abilitata, vengono registrati anche i documenti recuperati dalla knowledge base (se RAG è abilitato) e i prompt, che possono contenere informazioni riservate.

# VPC

La soluzione offre due opzioni per la configurazione di Amazon VPC:

1. Lascia che la soluzione crei un Amazon VPC per te.
2. Gestione e utilizzo del proprio Amazon VPC da utilizzare all'interno della soluzione.

## Lascia che la soluzione crei un Amazon VPC per te

Se selezioni l'opzione per consentire alla soluzione di creare un Amazon VPC, per impostazione predefinita verrà implementata come architettura 2-AZ con un intervallo CIDR 10.10.0.0/20. Hai la possibilità di utilizzare Amazon VPC IP Address Manager (IPAM), con 1 sottorete pubblica e 1 sottorete privata in ogni zona. La soluzione crea gateway NAT in ciascuna delle sottoreti pubbliche

e configura le funzioni Lambda per crearli nelle sottoreti private. ENIs Inoltre, questa configurazione crea tabelle di routing e relative voci, gruppi di sicurezza e relative regole, rete ACLs, endpoint VPC (gateway ed endpoint di interfaccia).

## Gestire il proprio Amazon VPC

Quando distribuisci la soluzione con un Amazon VPC, hai la possibilità di utilizzare un Amazon VPC esistente nel tuo account e nella tua regione AWS. Ti consigliamo di rendere disponibile il tuo VPC in almeno due zone di disponibilità per garantire un'elevata disponibilità. Il tuo VPC deve inoltre avere i seguenti endpoint VPC e le relative policy IAM associate per le configurazioni di VPC e tabella di routing.

### Per un pannello di controllo di distribuzione Amazon VPC

1. Endpoint gateway per DynamoDB.
2. Endpoint gateway per S3.
3. Endpoint di interfaccia per. CloudWatch
4. Endpoint di interfaccia per AWS CloudFormation.

### Per un caso d'uso Amazon VPC

1. Endpoint gateway per DynamoDB.
2. Endpoint gateway per S3.
3. Endpoint di interfaccia per. CloudWatch
4. Endpoint di interfaccia per Systems Manager Parameter Store.

> **Note**
>
> La soluzione richiede `com.amazonaws.region.ssm` solo.

5. Endpoint di interfaccia per Amazon Bedrock (bedrock-runtime, agent-runtime,). bedrock-agent-runtime
6. Facoltativo: se la distribuzione utilizzerà Amazon Kendra come knowledge base, è necessario un endpoint di interfaccia per Amazon Kendra.
7. Facoltativo: se la distribuzione utilizzerà un LLM in Amazon Bedrock, è necessario un endpoint di interfaccia per Amazon Bedrock.

> **Note**
>
> La soluzione richiede solo. `com.amazonaws.region.bedrock-runtime`

8. Facoltativo: se la distribuzione utilizzerà Amazon SageMaker AI per LLM, è necessario un endpoint di interfaccia per Amazon SageMaker AI.

> **Note**
>
> La soluzione non eliminerà o modificherà la configurazione VPC quando si utilizza l'opzione di distribuzione Bring your own VPC. Tuttavia, eliminerà tutto VPCs ciò che viene creato dalla soluzione nell'opzione Crea un VPC per me. Per questo motivo, è necessario prestare attenzione quando si condivide un VPC gestito dalla soluzione tra stack/distribuzioni.
> Ad esempio, la distribuzione A utilizza l'opzione Crea un VPC per me. La distribuzione B utilizza Bring my own VPC utilizzando il VPC creato dalla distribuzione A. Se la distribuzione A viene eliminata prima della distribuzione B, la distribuzione B non funzionerà più perché il VPC è stato eliminato. Inoltre, poiché la distribuzione B utilizza le funzioni ENIs create da Lambda, l'eliminazione della distribuzione A potrebbe comportare errori e la conservazione delle risorse residue.

## Amazon CloudFront

Questa soluzione implementa una console Web ospitata in un bucket Amazon S3. Per contribuire a ridurre la latenza e migliorare la sicurezza, questa soluzione include una CloudFront distribuzione con un'identità di accesso di origine, ovvero un CloudFront utente che fornisce l'accesso pubblico ai contenuti del bucket del sito Web della soluzione. Per ulteriori informazioni, consulta Limitazione dell'accesso ai contenuti Amazon S3 utilizzando un'identità Origin Access nella  CloudFront Amazon Developer Guide.

> **Note**
>
> CloudFront prevede un limite di quota flessibile a livello di account di 20 policy di intestazione di risposta. Questa soluzione crea policy di intestazione di risposta personalizzate per scopi di sicurezza. Se disponi di più di 20 distribuzioni di Generative AI Application Builder su AWS

o relativi casi d'uso, le nuove distribuzioni potrebbero fallire a causa del raggiungimento del limite di quota.

Per risolvere questo problema, puoi richiedere un aumento della quota per la quota di Response Header Policies nella console AWS Service Quotas seguendo questi passaggi:

1. Apri la console AWS Service Quotas.
2. Nel riquadro di navigazione, selezionare Servizi AWS.
3. Cerca e seleziona Amazon CloudFront.
4. Scorri fino alla quota Response Header Policies e scegli Richiedi un aumento della quota.
5. Segui le istruzioni per richiedere un aumento del limite di quota per il tuo account AWS.

Aumentando la quota di Response Header Policies, puoi garantire che le nuove implementazioni di Generative AI Application Builder su AWS o i relativi casi d'uso non falliscano a causa del limite di quota.

# Quote

Le quote di servizio, anche denominate limiti, rappresentano il numero massimo di risorse di servizio o operazioni per l'account AWS.

## Quote per i servizi AWS in questa soluzione

Assicurati di disporre di una quota sufficiente per ciascuno dei servizi implementati in questa soluzione. Per ulteriori informazioni, consulta le quote dei servizi AWS.

Utilizza i seguenti link per accedere alla pagina relativa al servizio. Per visualizzare le quote di servizio per tutti i servizi AWS nella documentazione senza cambiare pagina, visualizza invece le informazioni nella pagina Endpoint e quote del servizio nel PDF.

## Quote Amazon Bedrock AgentCore

Per le implementazioni di Agent Builder, tieni presente le seguenti quote di servizio Amazon AgentCore Bedrock:

| Quota | Stati Uniti orientali (Virginia settentrionale) | Altre regioni |
|---|---|---|
| Carichi di lavoro Active Session per account | 1000 | 500 |
| Agenti totali per account | 1.000 | 1.000 |
| Versioni per account | 1.000 | 1.000 |

# Implementazione della soluzione

Questa soluzione utilizza CloudFormation modelli e stack AWS per automatizzarne l'implementazione. Il CloudFormation modello specifica le risorse AWS incluse in questa soluzione e le relative proprietà. Lo CloudFormation stack fornisce le risorse descritte nel modello.

## Panoramica del processo di distribuzione

Prima di lanciare la soluzione, esaminate i costi, l'architettura, la sicurezza e altre considerazioni discusse in questa guida.

> **Important**
>
> Se prevedi di utilizzare Amazon Bedrock, devi richiedere l'accesso ai modelli prima che siano disponibili per l'uso. Per ulteriori dettagli, consulta Model access nella Amazon Bedrock User Guide.

Tempo di implementazione: circa 10 minuti

Fase 1: Avviare lo stack di dashboard di distribuzione

Fase 2: Implementazione di un caso d'uso

Fase 3: Implementazione di un caso d'uso utilizzando la procedura guidata del dashboard di distribuzione

Fase 4: Configurazione post-implementazione

Facoltativamente, puoi distribuire i casi d'uso separatamente dalla soluzione, se preferisci non avere l'interfaccia utente del dashboard di Deployment o. APIs

- Implementazione di un caso d'uso Text autonomo
- Implementazione di un caso d'uso autonomo di Bedrock Agent

Puoi anche fornire una configurazione di chat DynamoDB.

> **Important**
>
> Questa soluzione invia metriche operative ad AWS (i «Dati») sull'utilizzo di questa soluzione. Utilizziamo questi dati per comprendere meglio come i clienti utilizzano questa soluzione e i servizi e i prodotti correlati. La raccolta di questi dati da parte di AWS è soggetta alla politica sulla privacy di AWS.

## CloudFormation Modello AWS

Puoi scaricare il CloudFormation modello per questa soluzione prima di distribuirla.


genera
ai-application-builder-on-aws.template: utilizza questo modello per avviare la soluzione e tutti i componenti associati. La configurazione predefinita distribuisce le soluzioni principali e di supporto disponibili nei servizi AWS in questa sezione delle soluzioni, ma puoi personalizzare il modello per soddisfare le tue esigenze specifiche.

> **Note**
>
> Le CloudFormation risorse AWS vengono create a partire da costrutti di AWS Cloud Development Kit (AWS CDK).

Questo CloudFormation modello AWS distribuisce Generative AI Application Builder su AWS nel cloud AWS.

## Fase 1: Avvia lo stack di dashboard di distribuzione

Segui le step-by-step istruzioni in questa sezione per configurare e distribuire la soluzione nel tuo account.

Tempo di implementazione: circa 10 minuti

1. Accedi alla Console di gestione AWS e seleziona il pulsante per avviare il `generative-ai-application-builder-on-aws.template` CloudFormation modello.

2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra regione AWS, utilizza il selettore della regione nella barra di navigazione della console.

   > **Note**
   >
   > Questa soluzione utilizza Amazon Kendra e Amazon Bedrock, che attualmente non sono disponibili in tutte le regioni AWS. Se utilizzi queste funzionalità, devi avviare questa soluzione in una regione AWS in cui questi servizi sono disponibili. Per la disponibilità più aggiornata per regione, consulta l'AWS Regional Services List.

3. Nella pagina Create stack, verifica che l'URL del modello corretto sia nella casella di testo URL Amazon S3 e scegli Avanti.
4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni dei caratteri di denominazione, consulta IAM e STS Limits nella AWS Identity and Access Management User Guide.
5. In Parametri, esamina i parametri per questo modello di soluzione e modificali se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Predefinita | Description |
| --- | --- | --- |
| Email dell'utente amministratore | No | L'indirizzo e-mail dell'utente amministratore che avrà accesso alla dashboard di distribuzione. Se fornito, verranno creati un gruppo e un utente Amazon Cognito con le autorizzazioni per distribuire e gestire i casi d'uso. Puoi anche utilizzare `placeholder@example.com` per creare il gruppo |

| Parametro | Predefinita | Description |
| --- | --- | --- |
| | | ma non l'utente. Per informazioni sulla configurazione del pool di utenti, fare riferimento alla sezione Configurazione manuale del pool di utenti. |
| VpcEnabled | No | Il dashboard di distribuzione deve essere distribuito all'interno di un VPC? |
| CreateNewVpc | No | Disponibile solo se VpcEnabledlo è. Yes Se il valore èYes, lo stack creerà il VPC e distribuirà la soluzione all'interno del VPC creato.<br><br>Se VpcEnabledè Yes ed CreateNewVpcèNo, allora devi fornire una configurazione VPC esistente (ExistingVpcId,, ExistingPrivateSubnetIdsExistingSecurityGroupIds, VpcAzs). |
| IPAMPoolId | (Input opzionale) | È possibile configurare IPAM e fornire l'ID creato come input per assegnare l'intervallo di indirizzi IP che deve utilizzare la distribuzione di questo stack. Per i dettagli sull'IPAM, consulta Come funziona IPAM. |

| Parametro | Predefinita | Description |
| --- | --- | --- |
| Implementa UI | `Yes` | Hai la possibilità di implement are la dashboard di distribuz ione senza l'interfaccia utente Web (e le risorse AWS necessarie per la distribuz ione Web). In tal caso, la soluzione distribuirà tutta l'infrastruttura, inclusi gli endpoint dell'API REST. Questa opzione è utile per integrare la propria interfacc ia web con la dashboard di Deployment. APIs |
| ExistingVpcId | (Input opzionale) | Necessario solo se desideri implementare la soluzione in un VPC esistente che hai creato. |
| ExistingPrivateSubnetIds | (Input opzionale) | Necessario solo se desideri implementare la soluzione in un VPC esistente che hai creato. Le funzioni Lambda verranno distribuite in questa sottorete. |
| ExistingSecurityGroupIds | (Input opzionale) | Necessario solo se desideri implementare la soluzione in un VPC esistente che hai creato. Assicurati che i gruppi di sicurezza dispongano delle autorizzazioni per una connessione TCP in uscita. |

| Parametro | Predefinita | Description |
| --- | --- | --- |
| VpcAzs | (Input opzionale) | Necessario solo se desideri implementare la soluzione in un VPC esistente che hai creato. |
| CognitoDomainPrefix | (Input opzionale) | Richiesto solo se desideri distribuire la soluzione in un pool di utenti Amazon Cognito esistente che hai creato. Se non fornisci un valore, la soluzione lo genera. |
| ExistingCognitoUserPoolId | (Input opzionale) | Richiesto solo se desideri distribuire la soluzione in un pool di utenti Amazon Cognito esistente che hai creato. |
| ExistingCognitoUserPoolClient | (Input opzionale) | Richiesto solo se desideri distribuire la soluzione in un pool di utenti Amazon Cognito esistente che hai creato. Se non fornisci un valore, la soluzione crea un client con pool di utenti. Questo parametro può essere fornito solo se si fornisce un ExistingCognitoUserPoolIdvalore. |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi e crea, rivedi e conferma le impostazioni. Seleziona la casella per confermare che il modello creerà risorse AWS Identity and Access Management (IAM).
9. Scegli Invia per distribuire lo stack.

Puoi visualizzare lo stato dello stack nella CloudFormation console AWS nella colonna Status. Dovresti ricevere lo status CREATE_COMPLETE in circa 10 minuti.

# Fase 2: Implementazione di un caso d'uso

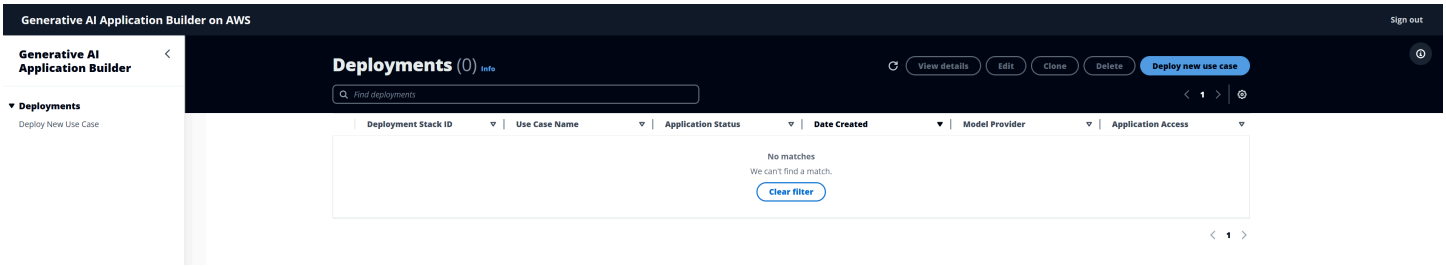
> **Important**
>
> Una volta che lo stack è stato distribuito correttamente, viene inviata un'e-mail di registrazione all'indirizzo e-mail dell'utente amministratore configurato. Utilizzando tali credenziali, l'utente amministratore può accedere alla dashboard di Deployment per utilizzare l'applicazione web.

> **Note**
>
> L' DevOps utente con accesso alla Console di gestione AWS deve fornire all'utente amministratore l' CloudFront URL dell'interfaccia utente del dashboard di distribuzione al termine dello stack. L'URL è disponibile nella scheda Outputs dello stack. CloudFormation

1. Accedi alla dashboard di Deployment come utente amministratore.
2. Nella pagina iniziale dell'applicazione, scegli Deploy new use case.

   Verrà avviata la procedura guidata di implementazione, che ti guida nella creazione dello use case.

Rappresenta la landing page della dashboard di Deployment: nuova implementazione



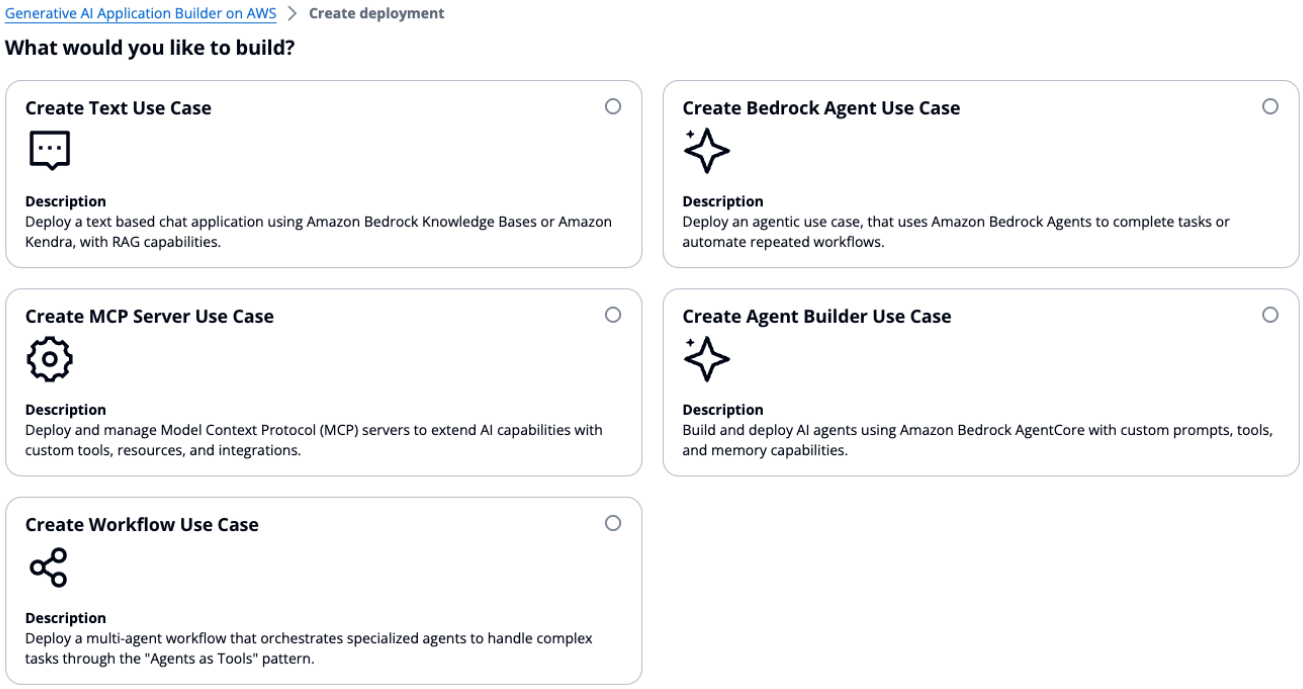

> **Note**
>
> Se devi aggiungere altri utenti alla tua implementazione, consulta la sezione Gestione del pool di utenti di Cognito per maggiori dettagli.

# Passaggio 3: implementa un caso d'uso utilizzando la procedura guidata del dashboard di distribuzione

Nella procedura guidata del dashboard di distribuzione, devi scegliere tra le seguenti opzioni:

- Caso d'uso testuale: distribuisce un'applicazione di chat, con funzionalità RAG opzionali
- Caso d'uso di Bedrock Agent: utilizza Amazon Bedrock Agents per completare attività o automatizzare flussi di lavoro ripetuti
- Server MCP: distribuisci e gestisci i server MCP con metodi gateway o di runtime
- Agent Builder: crea e distribuisci agenti personalizzati AgentCore con l'integrazione MCP e la gestione della memoria
- Workflow Builder - Orchestra più agenti Agent Builder utilizzando la delega gerarchica

Mostra cinque opzioni: Create Text use Case, Create Bedrock Agent Use Case, Create MCP Server Use Case, Create Agent Builder Use Case o Create Workflow Use Case.




## Fase 3a: Implementazione di un caso d'uso testuale

Questa sezione fornisce istruzioni per la distribuzione di un caso d'uso di tipo Text.

## Seleziona il caso d'uso

Quando scegli il caso d'uso Crea testo, l'interfaccia utente apre la schermata Seleziona caso d'uso. Inserisci le informazioni che seguono:

- Usa il nome del caso.
- Indirizzo e-mail opzionale per l'utente predefinito del caso d'uso da aggiungere al pool di utenti di Amazon Cognito per il caso d'uso e a cui assegnare le autorizzazioni per interagire con esso.
- Se desideri implementare un'interfaccia utente con questo caso d'uso. Se non desideri implementare un'interfaccia utente con lo use case, puoi utilizzare gli endpoint API distribuiti per utilizzarli con la tua applicazione.

## Dettagli dei casi d’uso

La fase relativa ai dettagli del caso d'uso consente di configurare impostazioni aggiuntive per la distribuzione.

Per impostazione predefinita, lo use case Text crea e configura un pool di utenti Amazon Cognito per te quando la soluzione implementa la dashboard di distribuzione. La soluzione autentica nuovi casi d'uso con un client appena creato nello stesso pool di utenti. Tuttavia, puoi fornire un ID del pool di utenti e un ID client esistenti in questa fase se desideri utilizzare il tuo pool di utenti e client Amazon Cognito con lo use case.

**Important**

Gli utenti amministratori hanno accesso a tutti i casi d'uso distribuiti quando il pool di utenti di Amazon Cognito viene creato tramite la procedura guidata di distribuzione. Se fornisci il tuo pool di utenti durante la distribuzione, devi assicurarti che l'amministratore disponga delle autorizzazioni per accedere ai casi d'uso distribuiti.
Dovrai inoltre aggiornare la richiamata consentita URLs e la disconnessione consentita nei client dell'app URLs in Cognito. Per farlo:

1. Passa alla console Cognito
2. Scegli User Pools (Pool di utenti).
3. Scegli il tuo pool di utenti.
4. Scegli App Clients nel menu a sinistra.
5. Scegli il client dell'app che desideri modificare.

6. Scegli la scheda Pagine di accesso.
7. Scegli Modifica e aggiungi il tuo URLs.
8. Scegli Save changes (Salva modifiche).

Inoltre, se devi aggiungere altri utenti a un caso d'uso, consulta la sezione Gestione del pool di utenti di Cognito.

## Seleziona la configurazione di rete

Questa procedura guidata consente di implementare lo use case con un Amazon Virtual Private Cloud (Amazon VPC) preesistente o nuovo. Se si seleziona un VPC preesistente, è necessario fornire un ID VPC, fino a 16 ID di sottorete e fino a 5 gruppi di sicurezza da IDs utilizzare con questo VPC. Se non utilizzi un VPC preesistente, queste impostazioni verranno configurate automaticamente.

## Selezione del modello

Nella fase Seleziona il modello, puoi scegliere il fornitore del modello dal menu a discesa. Sono disponibili due opzioni: Bedrock e. SageMaker

Se si seleziona SageMaker, è possibile creare un endpoint del modello SageMaker AI nella console SageMaker AI e fornire lo schema di input previsto dal modello e l'output JSONPath per la risposta LLM. Puoi fare riferimento alla sezione Using Amazon SageMaker AI as an LLM Provider e agli esempi di payload SageMaker AI forniti nel repository della GitHub soluzione.

Se selezioni Amazon Bedrock, ti verranno presentate quattro opzioni:

- Modelli Quick Start: inizia rapidamente con una raccolta di modelli con price/performance caratteristiche diverse. Consigliato per creare le tue prime app. Questa opzione consente di selezionare il nome di un modello dall'elenco fornito.
- Altri modelli Foundation: accedi alla gamma completa di modelli di fondazione con diverse capacità e specializzazioni. Questa opzione consente di inserire l'ID del modello per il modello di fondazione Bedrock on-demand desiderato.
- Profili di inferenza: i profili di inferenza sfruttano l'inferenza interregionale di Bedrock per aumentare la velocità effettiva e migliorare la resilienza instradando le richieste su più regioni AWS durante i picchi di utilizzo. Questa opzione ti consente di inserire l'ID del profilo di inferenza che desideri utilizzare.

- Modelli forniti: capacità di throughput dedicata per carichi di lavoro di produzione che richiedono prestazioni costanti. Questa opzione ti consente di inserire l'ARN del provisioned/custom modello da utilizzare da Amazon Bedrock.

La fase di selezione del modello consente inoltre di scegliere le impostazioni avanzate del modello. Consulta le impostazioni Advanced LLM per dettagli sulla configurazione di Amazon Bedrock Guardrails, sul throughput assegnato per Amazon Bedrock e sui parametri aggiuntivi del modello.

Inferenza tra regioni

L'inferenza tra regioni aiuta gli utenti di Amazon Bedrock a gestire senza problemi i picchi di traffico non pianificati utilizzando l'elaborazione in diverse regioni AWS. Per utilizzare l'inferenza tra regioni, è necessario il profilo di inferenza. Un profilo di inferenza è un'astrazione su un pool di risorse su richiesta da un set configurato di regioni AWS. Può indirizzare la richiesta di inferenza, proveniente dalla regione di origine, verso un'altra regione configurata in quel pool. Ciò consente la distribuzione del traffico su più regioni AWS. Questo aiuta a consentire un throughput più elevato e una maggiore resilienza durante i periodi di picco della domanda.

I profili di inferenza prendono il nome dal modello e dalle regioni che supportano. È necessario richiamare un profilo di inferenza da una delle regioni che include. Ad esempio, come illustrato nella tabella seguente, l'ID del profilo di inferenza `us.anthropic.claude-3-haiku-20240307-v1:0` consente la distribuzione del traffico tra `us-east-1` le `us-west-2` regioni del modello scelto. Alcuni modelli sono disponibili solo con un profilo di inferenza in una particolare regione.

| Profilo di inferenza | ID del profilo di inferenza | Regioni incluse |
|---|---|---|
| US Anthropic Claude 3 Haiku | `us.anthropic.claude-3-haiku-20240307-v1:0` | Stati Uniti orientali (Virginia settentrionale) (`us-east-1` )<br><br>Stati Uniti occidentali (Oregon) (`us-west-2` ) |

Se desideri utilizzare un ID del profilo di inferenza anziché un ID del modello, devi identificare l'ID del profilo di inferenza appropriato. Per ulteriori informazioni, consulta Regioni e modelli supportati per i profili di inferenza nella Amazon Bedrock User Guide. Nella console Amazon Bedrock, l'opzione di inferenza interregionale nel menu di navigazione a sinistra fornisce questi profili di inferenza. IDs

Dopo aver identificato l'ID del profilo di inferenza da utilizzare, puoi utilizzarlo durante la fase di selezione del modello eseguendo le seguenti operazioni:

1. Seleziona Amazon Bedrock come fornitore del modello.
2. Seleziona l'opzione del pulsante radio Inference Profiles.
3. Inserisci l'ID del tuo profilo di inferenza nella casella di testo visualizzata.

Per ulteriori dettagli sui profili di inferenza, consulta Improve resilience with cross-region inference nella Amazon Bedrock User Guide.

## Seleziona la knowledge base

Se stai cercando di implementare un caso d'uso diverso da Retrieval Augmented Generation (RAG), puoi saltare questo passaggio.

Tuttavia, se desideri abilitare RAG come parte della tua distribuzione, ora puoi fornire un Amazon Kendra Index Id preconfigurato o un Amazon Bedrock Knowledge Base ID. Puoi anche creare un nuovo Amazon Kendra Index da utilizzare con la soluzione. La soluzione attualmente supporta Amazon Kendra e Amazon Bedrock Knowledge Base come knowledge base per la distribuzione di use case basati su RAG.

Consulta la sezione Configurazione di una Knowledge Base per le linee guida sull'inserimento di dati nella knowledge base da utilizzare con la distribuzione basata su RAG.

Configurazioni RAG avanzate

La procedura guidata consente di selezionare opzioni avanzate da utilizzare con l'implementazione RAG, ad esempio il numero di documenti da recuperare ogni volta che viene inviata una query alla knowledge base, una risposta testuale statica dal LLM quando non viene trovato alcun documento nella knowledge base, se si desidera visualizzare le fonti dei documenti con la risposta LLM per i controlli di integrità, ecc. Puoi inoltre configurare configurazioni specifiche della knowledge base per Amazon Kendra, ad esempio Role-based Access Control (RBAC) o Override Search Type quando usi Amazon Serverless con Amazon Bedrock Knowledge Bases. OpenSearch Consulta la sezione Impostazioni avanzate della Knowledge Base per maggiori dettagli su queste impostazioni avanzate.

> **Note**
>
> La Knowledge Base deve trovarsi nello stesso account e nella stessa regione degli stack di dashboard di Deployment e case case distribuiti.

Seleziona i prompt e i limiti dei token

In questo passaggio, puoi configurare il prompt per l'utilizzo con l'LLM. I prompt possono richiedere segnaposti come, e. `{input}` `{history}` `{context}` Questi segnaposto indicano all'LLM da dove attingere l'input dell'utente, la cronologia delle conversazioni e le informazioni recuperate dalla knowledge base.

- Per il fornitore di modelli Bedrock, è necessario fornire il prompt di sistema, che non presenta restrizioni per un caso d'uso diverso da RAG. La richiesta di chiarimento delle ambiguità per il fornitore di modelli Bedrock richiede tuttavia un minimo di due segnaposto e `{input}` `{history}`
- Per quanto riguarda il fornitore SageMaker del modello, il sistema e le istruzioni di disambiguazione, entrambi richiedono un minimo di due segnaposto: e. `{input}` `{history}`
- Per i casi d'uso RAG, per ogni fornitore di modelli, è richiesto in aggiunta il segnaposto. `{context}`

Per ulteriori informazioni, consulta Configurazione delle istruzioni. Puoi anche fare riferimento alla sezione Suggerimenti per la gestione dei limiti dei token del modello mentre selezioni le dimensioni dei limiti dei token per i tuoi prompt.

Abilita l'input multimodale

Questo passaggio consente di abilitare le funzionalità di input multimodali per il proprio caso d'uso. Se abilitata, gli utenti possono caricare e inviare immagini e documenti insieme alle loro query di testo.

Tipi di file e vincoli supportati:

- Immagini: fino a 20 immagini per messaggio. Ogni immagine non deve avere più di 3,75 MB di dimensione e 8.000 px di altezza e larghezza. Formati supportati: png, jpeg, gif, webp
- Documenti: fino a 5 documenti per messaggio. Ogni documento non deve avere una dimensione superiore a 4,5 MB. Formati supportati: pdf, csv, doc, docx, xls, xlsx, html, txt, md

Come usare l'input multimodale:

1. Abilita il MultimodalEnabledparametro durante la distribuzione dei casi d'uso
2. Nell'interfaccia di chat, gli utenti possono caricare file in due modi:
   - Facendo clic sul pulsante di caricamento nella casella di immissione della chat, oppure

- Trascinare e rilasciare i file direttamente nell'interfaccia della chat

3. I file vengono caricati su Amazon S3 ed elaborati dal modello selezionato
4. I file caricati vengono eliminati automaticamente dopo 48 ore

Monitoraggio dello stato dei file:

DevOps gli utenti possono monitorare i metadati dei file in DynamoDB, che includono il tempo di caricamento e lo stato di elaborazione. I file possono avere i seguenti stati:

- in sospeso: il caricamento del file è stato avviato ma non ancora completato. Questo è lo stato iniziale quando viene generato un URL predefinito.
- caricato - Il file è stato caricato con successo su S3 ed è pronto per essere elaborato dal modello.
- cancellato: il file è stato eliminato dall'utente e non dovrebbe più essere accessibile per l'elaborazione.
- non valido: controlli di convalida del file non riusciti (ad esempio, mancata corrispondenza del tipo di file o errore di convalida di sicurezza).

I file in sospeso che non vengono mai caricati verranno ripuliti automaticamente alla scadenza del TTL. Solo i file con lo stato di caricamento possono essere elaborati dal modello.

Il bucket multimodale S3 e la tabella di metadati DynamoDB sono disponibili negli output del Deployment Dashboard con le chiavi e, rispettivamente. `MultimodalDataBucketName` `MultimodalDataMetadataTable`

> **Note**
>
> Non tutti i modelli supportano l'input multimodale. Assicurati che il modello selezionato supporti l'elaborazione di immagini e documenti prima di attivare questa funzione. Consulta i modelli di base supportati nella documentazione di Amazon Bedrock per verificare quale modello supporta Image come modalità di input.

> **Important**
>
> I file caricati dagli utenti vengono archiviati in Amazon S3 con una politica del ciclo di vita di 48 ore. I metadati sui file caricati vengono archiviati in Amazon DynamoDB con un TTL di 24 ore per la cronologia delle conversazioni.

#### Revisione e implementazione

Dopo questo passaggio, rivedi le impostazioni selezionate e scegli Deploy Use Case. Il nuovo use case viene quindi implementato e diventa visibile nella visualizzazione del dashboard di Deployment per gestirlo ulteriormente.

### Fase 3b: Implementazione di un caso d'uso di Bedrock Agent

Lo use case Bedrock Agent fornisce un meccanismo potente e sicuro per richiamare Amazon Bedrock Agents nei tuoi casi d'uso. Questa funzionalità consente agli sviluppatori di integrare senza problemi le funzionalità degli agenti autonomi basati sull'intelligenza artificiale in grado di orchestrare ed eseguire attività in più fasi su vari modelli di base, fonti di dati, applicazioni software e conversazioni con gli utenti, mantenendo al contempo solide misure di sicurezza.

#### Prerequisiti

Prima di creare un agente Amazon Bedrock, assicurati di disporre di quanto segue:

1. L'account AWS in cui è distribuito Generative AI Application Builder su AWS, con accesso alla console Amazon Bedrock.
2. Autorizzazioni IAM appropriate per creare e gestire Amazon Bedrock Agents.

#### Creazione di un agente Amazon Bedrock

Per istruzioni dettagliate sulla creazione di un agente, consulta la sezione Crea e configura l'agente manualmente nella Amazon Bedrock User Guide. Puoi configurare opzioni come:

- Istruzioni (prompt) per il tuo agente
- Knowledge base, utilizzata per cercare informazioni aggiuntive in base all'input dell'utente
- Memoria dell'agente per consentire agli agenti di ricordare le informazioni in più sessioni (per un massimo di 30 giorni)

Dopo aver creato con successo un agente Amazon Bedrock, puoi procedere al flusso guidato dei casi d'uso di Generative AI Application Builder su AWS Bedrock Agent. Per farlo, scegli Deploy a new use case nella dashboard di Deployment e seleziona Create Bedrock Agent Use Case. Segui la procedura guidata e utilizza i seguenti passaggi per configurare lo use case.

Seleziona il caso d'uso

Questo passaggio è lo stesso del caso d'uso Text descritto in precedenza.

Seleziona la configurazione di rete

Questo passaggio è lo stesso del caso d'uso Text descritto in precedenza

Seleziona agente

In questa fase, devi fornire l'ID agente e l'ID alias dell'agente Amazon Bedrock che hai creato.

## Fase 3c: Implementazione di un caso d'uso del server MCP

Lo use case del server MCP (Model Context Protocol) consente di distribuire e gestire server MCP che possono essere integrati con modelli e agenti AI. I server MCP forniscono un modo standardizzato per esporre strumenti, risorse e funzionalità alle applicazioni AI. È possibile creare server MCP a partire da funzioni APIs Lambda esistenti oppure ospitare server MCP personalizzati utilizzando immagini di container.

Prerequisiti

Prima di implementare un caso d'uso del server MCP, assicuratevi di disporre di quanto segue:

1. L'account AWS in cui viene distribuito Generative AI Application Builder su AWS.
2. Autorizzazioni IAM appropriate per creare e gestire risorse Amazon Bedrock AgentCore .
3. A seconda del metodo di creazione scelto:
   - Per il metodo Gateway (Lambda/API/MCPServer): funzioni Lambda, endpoint API con i file di schema corrispondenti (formato JSON per Lambda, OpenAPI/Smithy for APIs) o endpoint URL del server MCP
   - Per il metodo Runtime (ECR): un'immagine del contenitore Docker inviata ad Amazon ECR contenente l'implementazione del server MCP

Metodi di creazione del server MCP

La soluzione supporta due metodi per la creazione di server MCP:

Crea da Lambda, API o MCP Server (metodo Gateway)

Questo metodo crea un gateway MCP che include funzioni Lambda, REST o server MCP esterni esistenti APIs, rendendoli accessibili come strumenti MCP. Il gateway gestisce la traduzione del protocollo tra MCP e i servizi esistenti.

- Obiettivi Lambda: integra le funzioni Lambda esistenti fornendo la funzione ARN e un file di schema JSON che descrive il formato della funzione input/output
- Obiettivi OpenAPI: integra REST utilizzando le specifiche APIs OpenAPI (formato JSON o YAML) con supporto per l'autenticazione 2.0 o API Key OAuth
- Obiettivi Smithy: integrazione APIs definita utilizzando i file del modello Smithy (formato.smithy o.json)
- Obiettivi del server MCP: Connettiti direttamente a server MCP esterni tramite endpoint URL, consentendo l'integrazione dei server MCP esistenti senza implementare una nuova infrastruttura

È possibile configurare più destinazioni (fino a 10) all'interno di un singolo gateway MCP, ognuna delle quali rappresenta uno strumento o una funzionalità diversi.

Hosting da ECR Image (metodo Runtime)

Questo metodo implementa un server MCP containerizzato da un'immagine Amazon ECR. Utilizza questo approccio quando disponi di un'implementazione server MCP personalizzata che deve essere eseguita come servizio autonomo.

- Fornite l'URI dell'immagine ECR (deve includere un tag, ad esempio, o) `:latest` `:v1.0.0`
- Facoltativamente, configura le variabili di ambiente per passare la configurazione al tuo contenitore
- Il contenitore deve implementare il protocollo MCP ed esporre gli endpoint richiesti

Implementazione di un server MCP

Per distribuire un caso d'uso del server MCP, scegli Implementa un nuovo caso d'uso nella dashboard di distribuzione e seleziona Crea caso d'uso del server MCP. Segui la procedura guidata e utilizza i seguenti passaggi per configurare lo use case.

Seleziona il caso d'uso

Questo passaggio è lo stesso del caso d'uso Text descritto in precedenza.

Seleziona la configurazione di rete

Attualmente è abilitato solo l'accesso pubblico e il VPC non è supportato per la configurazione di rete.

Crea un server MCP

In questo passaggio, configuri la distribuzione del server MCP:

Metodo di creazione del server MCP

Scegliete tra i due metodi di creazione:

- Crea da Lambda, API o MCP Server: crea un gateway MCP da funzioni Lambda, specifiche API o endpoint server MCP esterni esistenti
- Hosting da un'immagine ECR: implementa un server MCP personalizzato da un'immagine del contenitore

> **Note**
>
> Il metodo di creazione non può essere modificato dopo la distribuzione. Se è necessario cambiare metodo, è necessario implementare un nuovo use case del server MCP.

Configurazione del gateway (per il metodo Lambda/API/MCP Server)

Se hai selezionato il metodo Gateway, configura una o più destinazioni:

1. Nome della destinazione (obbligatorio): un nome descrittivo per identificare questa configurazione di destinazione
2. Descrizione dell'obiettivo (opzionale): una breve descrizione di ciò che fa questo obiettivo
3. Tipo di destinazione: seleziona il tipo di oggetto da configurare:
   - Lambda: per le funzioni AWS Lambda
   - OpenAPI: per REST con specifiche APIs OpenAPI
   - Smithy: Per APIs le definizioni dei modelli Smithy
   - Server MCP: per la connessione diretta a server MCP esterni tramite endpoint URL
4. File di schema (obbligatorio): carica il file di schema che descrive il tuo obiettivo:

- Per Lambda: file di schema JSON che descrive il formato. input/output Per i dettagli sulla creazione di schemi di strumenti Lambda, consulta lo schema degli strumenti Lambda nella Amazon Bedrock Developer Guide. AgentCore
- Per OpenAPI: file delle specifiche OpenAPI (JSON o YAML). Per informazioni dettagliate sui requisiti dello schema OpenAPI, consulta lo schema OpenAPI nella Amazon Bedrock Developer Guide. AgentCore
- Per Smithy: file modello Smithy (.smithy or .json). Per informazioni dettagliate sulla creazione degli obiettivi Smithy, consulta Building Smithy nella Amazon Bedrock Developer Guide. AgentCore

5. ARN della funzione Lambda (richiesto per i target Lambda): l'ARN della funzione Lambda da integrare
6. URL del server MCP (obbligatorio per le destinazioni del server MCP): l'endpoint URL del server MCP esterno a cui connettersi. L'URL deve essere codificato correttamente e il server MCP deve supportare le funzionalità degli strumenti con le versioni del protocollo MCP 2025-06-18. Per ulteriori informazioni, consulta gli obiettivi dei server MCP nella Amazon Bedrock AgentCore Developer Guide.
7. Autenticazione in uscita (richiesta per i target OpenAPI): configura l'autenticazione per le chiamate API REST:
   - Tipo di autenticazione: scegli OAuth 2.0 o chiave API
   - ARN del provider di autenticazione in uscita: l'ARN del provider di credenziali nel vault di token Amazon Bedrock AgentCore
   - Configurazioni aggiuntive: a seconda del tipo di autenticazione:
     - Per OAuth 2.0: configura ambiti e parametri personalizzati
     - Per la chiave API: specifica la posizione (parametro di intestazione o query), il nome del parametro e il prefisso opzionale

Puoi aggiungere più destinazioni (fino a 10) scegliendo Aggiungi un'altra destinazione. Ogni destinazione rappresenta uno strumento o una funzionalità separata esposta dal server MCP.

Configurazione ECR (per il metodo ECR Image)

Se hai selezionato il metodo Runtime, fornisci:

1. URI dell'immagine ECR (richiesto): l'URI completo dell'immagine Docker in Amazon ECR
   - Formato: `account-id.dkr.ecr.region.amazonaws.com/repository-name:tag`

- L'immagine deve trovarsi nella stessa regione AWS della distribuzione
- È richiesto un tag (ad esempio`:latest`,`:v1.0.0`)

2. Variabili di ambiente (opzionali): configura le coppie chiave-valore da passare al contenitore in fase di esecuzione
   - Utilizzale per fornire configurazione, credenziali o flag personalizzati
   - Puoi aggiungere fino a 10 variabili di ambiente

Revisione e implementazione

Dopo aver configurato il server MCP, rivedi le impostazioni selezionate e scegli Deploy Use Case. Il nuovo use case MCP Server viene quindi distribuito e diventa visibile nella visualizzazione del dashboard di Deployment per un'ulteriore gestione.

> **Note**
>
> Le implementazioni di MCP Server creano risorse in Amazon Bedrock AgentCore, inclusi gateway, runtime e identità dei carichi di lavoro. Queste risorse vengono gestite automaticamente dalla soluzione e verranno ripulite quando elimini lo use case.

## Fase 3d: Implementazione di un caso d'uso di Agent Builder

Agent Builder ti consente di creare, configurare e distribuire agenti AI pronti per la produzione su Amazon Bedrock. AgentCore Questa funzionalità offre il pieno controllo sul comportamento degli agenti tramite istruzioni di sistema, selezione del modello, integrazione del server MCP e gestione della memoria.

Il processo di distribuzione è principalmente lo stesso di un caso d'uso Text, con alcune differenze notevoli.

Seleziona il caso d'uso

Questo passaggio è lo stesso del caso d'uso Text descritto in precedenza.

Dettagli dei casi d’uso

Questo passaggio è lo stesso del caso d'uso Text descritto in precedenza.

### Configurare l'agente

In questo passaggio, configuri le impostazioni principali dell'agente, tra cui il prompt di sistema, servers/Strands gli strumenti MCP disponibili e la memoria.

#### Prompt di sistema

Il prompt di sistema definisce il comportamento, la personalità e le capacità dell'agente. Puoi:

- Modifica il modello di prompt di sistema predefinito
- Utilizzate il pulsante Ripristina i valori predefiniti per ripristinare il modello originale
- Includi istruzioni per l'utilizzo dello strumento e la formattazione delle risposte

#### Integrazione con server MCP (opzionale)

Configura i server Model Context Protocol per fornire al tuo agente l'accesso a strumenti e dati aziendali:

1. Seleziona uno dei server MCP disponibili nel menu a discesa
2. Consulta gli strumenti pronti all'uso disponibili che saranno accessibili all'agente

> **Note**
>
> I server MCP devono essere configurati e accessibili prima della distribuzione. Fate riferimento alla documentazione MCP per le istruzioni di configurazione del server.

#### Configurazione della memoria

Configura il modo in cui l'agente mantiene il contesto e le conoscenze:

- Memoria a breve termine: abilitata per impostazione predefinita per tutti gli agenti. Mantiene il contesto della conversazione all'interno delle sessioni.
- Memoria a lungo termine: attiva questa opzione per abilitare l'estrazione e l'archiviazione delle informazioni tra le sessioni. Utilizza la AgentCore memoria con una strategia di memoria semantica.

### Revisione e implementazione

Dopo questo passaggio, rivedi le impostazioni selezionate e scegli Deploy Use Case. L'implementazione di Agent Builder viene in genere completata in 10-15 minuti. Il nuovo caso d'uso diventa quindi visibile nella visualizzazione del dashboard di Deployment per essere ulteriormente gestito.

## Fase 3e: Implementazione di un caso d'uso di Workflow

Workflow Builder consente di creare agenti supervisori che orchestrano più agenti Agent Builder utilizzando il modello di delega Agents as Tools. Questa funzionalità consente di creare flussi di lavoro multiagente complessi riutilizzando le distribuzioni esistenti di Agent Builder.

Il processo di distribuzione segue uno schema simile a quello di Agent Builder, con passaggi aggiuntivi per l'individuazione e la selezione degli agenti.

### Seleziona il caso d'uso

Questo passaggio è lo stesso del caso d'uso Text descritto in precedenza.

### Dettagli dei casi d’uso

Questo passaggio è lo stesso del caso d'uso Text descritto in precedenza.

### Configura l'agente supervisore

In questo passaggio, si configura l'agente supervisore che coordinerà gli agenti specializzati di Agent Builder.

### Prompt di sistema

Il prompt di sistema definisce il modo in cui i delegati dell'agente supervisore lavorano agli agenti specializzati. Puoi:

- Modifica il modello di prompt di sistema predefinito
- Include istruzioni per la selezione e la delega degli agenti
- Definisci come aggregare i risultati di più agenti
- Utilizzate il pulsante Ripristina i valori predefiniti per ripristinare il modello originale

> **Note**
>
> Il prompt di sistema dovrebbe descrivere chiaramente quando e come utilizzare ciascun agente specializzato. Le descrizioni degli agenti sono fondamentali per una corretta delega.

#### Selezione del modello

Seleziona il modello di base per l'agente supervisore. L'agente supervisore utilizza questo modello per:

- Comprendere le richieste degli utenti
- Seleziona gli agenti specializzati appropriati
- Coordinare l'esecuzione degli agenti
- Aggrega e formatta le risposte

#### Seleziona agenti specializzati

In questo passaggio, si selezionano gli agenti di Agent Builder a cui il supervisore può delegare il lavoro.

#### Aggiungere agenti

1. Fate clic su Aggiungi agente per aprire la finestra di dialogo di selezione dell'agente
2. Seleziona uno o più agenti Agent Builder dall'elenco
3. Consulta le descrizioni degli agenti che verranno fornite al supervisore
4. Conferma la selezione

> **Note**
>
> - I flussi di lavoro richiedono almeno 1 caso d'uso di Agent Builder come agente specializzato
> - Tutti gli agenti specializzati devono essere implementati correttamente prima di creare il flusso di lavoro

Revisione e implementazione

Rivedi la configurazione del flusso di lavoro, tra cui:

- Prompt e modello del sistema dell'agente supervisore
- Elenco degli agenti specializzati
- Impostazioni della memoria

Scegli Deploy Use Case. L'implementazione di Workflow viene in genere completata in 15-20 minuti. Il nuovo flusso di lavoro diventa visibile nella visualizzazione del dashboard di Deployment per gestirlo ulteriormente.

# Fase 4: Configurazione post-implementazione

Questa sezione fornisce consigli per configurare la soluzione dopo la distribuzione.

## Versionamento dei bucket Amazon S3, politiche del ciclo di vita e replica tra regioni

Questa soluzione non impone configurazioni del ciclo di vita sui bucket che crea. Consigliamo quanto segue:

- Impostazione delle configurazioni del ciclo di vita per le implementazioni di produzione. Per maggiori dettagli, consulta la sezione Impostazione della configurazione del ciclo di vita su un bucket nella Guida per l'utente di Amazon Simple Storage Service.
- Abilitazione del controllo delle versioni e della replica tra regioni per i bucket Amazon S3 in base allo use case per il quale viene distribuita la soluzione.

## Backup di Amazon DynamoDB

Questa soluzione utilizza DynamoDB per diversi scopi (vedi i servizi AWS in questa soluzione). La soluzione non abilita i backup per le tabelle che crea. Consigliamo di creare un backup di questa funzionalità per le distribuzioni di produzione. Per ulteriori informazioni, consulta Backup di una tabella DynamoDB e Utilizzo di AWS Backup for DynamoDB.

## CloudWatch Dashboard e allarmi Amazon

La soluzione implementa una dashboard personalizzata per eseguire il rendering CloudWatch di grafici da metriche pubblicate personalizzate e metriche dei servizi AWS. Consigliamo di creare CloudWatch allarmi e aggiungere notifiche in base al caso d'uso per il quale viene implementata la soluzione.

## CloudWatch Registri Amazon

I log Lambda sono configurati per non scadere mai e i log di API Gateway sono configurati con una scadenza di 10 anni. È possibile aggiornare la scadenza dei rispettivi gruppi di log per allinearla alla politica di conservazione dei record dell'azienda.

## Domini web personalizzati con certificati TLS v1.2 o versioni successive

La soluzione implementa un'interfaccia utente Web e un gateway API ottimizzato per Edge utilizzando CloudFront. CloudFrontil dominio non applica i certificati TLS v1.2 o versioni successive. Ti consigliamo di creare un dominio personalizzato utilizzando Amazon Route 53, di creare un certificato utilizzando AWS Certificate Manager o di utilizzare un certificato esistente se l'organizzazione ne ha uno.

Per ulteriori dettagli, consulta la Amazon Route 53 Developer Guide e la scelta di una versione TLS minima per un dominio personalizzato in API Gateway.

## Scalabilità con Amazon Kendra

Questa soluzione offre la possibilità di utilizzare Amazon Kendra per eseguire ricerche intelligenti basate sulla tecnologia NLP tra i documenti acquisiti. Puoi aumentare la capacità di Amazon Kendra utilizzando i CloudFormation seguenti parametri per carichi di lavoro più grandi:

| Parametro | Predefinita | Description |
| --- | --- | --- |
| Capacità di interrogazione aggiuntiva di Amazon Kendra | `0` | La quantità di capacità di interrogazione aggiuntiva per un indice e GetQuerySuggestionsuna capacità. Un'unità di capacità aggiuntiva per un indice fornisce circa 8.000 query al giorno. |

| Parametro | Predefinita | Description |
| --- | --- | --- |
| Capacità di storage aggiuntiva di Amazon Kendra | `0` | Quantità di capacità di storage aggiuntiva per un indice. Un'unità a capacità singola offre 30 GB di spazio di archiviazione o 100.000 documenti, a seconda dell'evento che si verifica per primo. |
| Edizione Amazon Kendra | `Developer` | Amazon Kendra fornisce le edizioni Developer ed Enterprise per creare indici. Per ulteriori informazioni sulle differenze tra le edizioni di Amazon Kendra, consulta i prezzi di Amazon Kendra. |

Per modificare i valori di questi CloudFormation parametri, seleziona i valori appropriati al momento della distribuzione dello stack. Per ulteriori informazioni sulle unità di interrogazione e capacità di archiviazione, vedere Adjusting capacity.

> **Note**
>
> Se lo use case Text non viene distribuito con RAG abilitato, non viene utilizzato o creato un indice Amazon Kendra.

## Configurazione dell'SSO utilizzando la federazione Idp

Questa soluzione consente l'integrazione con provider di identità esterni che supportano la federazione delle identità basata su SAML o OIDC. Quando la soluzione viene implementata, crea un pool di utenti Amazon Cognito e l'integrazione di singoli client di app per la dashboard di distribuzione e i singoli casi d'uso. In base all'Idp esterno, segui i passaggi forniti nella sezione Configurazione dei provider di identità per il tuo pool di utenti della Amazon Cognito Developer Guide e scegli

l'integrazione del client dell'app per la dashboard di distribuzione o il caso d'uso con cui desideri configurare l'SSO.

Per trasferire le informazioni sul gruppo di utenti alla knowledge base o agli archivi vettoriali in un'architettura basata su RAG, dovrai mappare i gruppi di utenti dall'Idp esterno ai gruppi di utenti Amazon Cognito. La soluzione fornisce un trigger iniziale della funzione Lambda dello scaffolding da mappare con la fase precedente alla generazione del token. La funzione Lambda ha il file group_mapping.json che deve essere aggiornato per fornire le mappature del gruppo. Fai riferimento a Personalizzazione dei flussi di lavoro del pool di utenti con i trigger Lambda per i trigger Lambda supportati da Amazon Cognito.

## Configurazione manuale del pool di utenti

Se scegli di non inviare un indirizzo e-mail di amministratore o utente predefinito durante la distribuzione, devi creare manualmente i gruppi di utenti appropriati in Amazon Cognito per garantire le autorizzazioni corrette:

1. Per la dashboard Deployment, crea un gruppo denominato `Admin` nel tuo pool di utenti di Cognito.
2. Per ogni caso d'uso, crea un gruppo denominato `${UseCaseName}-Users` nel tuo pool di utenti di Cognito, dove si `${UseCaseName}` trova il nome del caso d'uso distribuito.

Questi gruppi sono necessari per il corretto funzionamento del meccanismo di autorizzazione. Tutti gli utenti a cui desideri concedere l'accesso devono essere aggiunti ai gruppi appropriati.

Se `placeholder@example.com` viene superato, verrà creato il gruppo Cognito, ma è comunque necessario creare gli utenti associati e assegnarli al gruppo.

## Personalizzazione della schermata di accesso

Questa soluzione utilizza l'interfaccia utente ospitata da Amazon Cognito per eseguire il rendering della pagina di accesso. Per personalizzare la pagina di accesso integrata, consulta Personalizzazione delle pagine Web di accesso e registrazione integrate nella Amazon Cognito Developer Guide.

## Ulteriori considerazioni sulla sicurezza

In base al caso d'uso per il quale distribuisci la soluzione, consulta i seguenti consigli di sicurezza:

- Chiavi di crittografia AWS KMS gestite dal cliente: la soluzione utilizza per impostazione predefinita chiavi AWS KMS gestite da AWS, poiché sono disponibili senza costi aggiuntivi. Esamina il tuo caso d'uso per determinare se è necessario aggiornare la soluzione per utilizzare chiavi AWS KMS gestite dal cliente.
- Regole di limitazione API Gateway: la soluzione viene implementata con regole di limitazione predefinite su API Gateway. In base al caso d'uso e ai volumi di transazioni previsti, ti consigliamo di configurare la limitazione per. APIs Per maggiori dettagli, consulta la sezione Richieste API Throttle per una migliore velocità effettiva nella Amazon API Gateway Developer Guide.
- Abilitazione di AWS CloudTrail: come pratica di sicurezza consigliata, prendi CloudTrail in considerazione l'abilitazione di AWS nell'account AWS in cui è distribuita la soluzione per registrare le chiamate API nell'account AWS. Per i dettagli, consulta la AWS CloudTrail User Guide.
- Rilevamento delle deviazioni: consigliamo di configurare il rilevamento della deriva sugli CloudFormation stack per identificare e ricevere notifiche in caso di modifiche involontarie o dannose allo stack di soluzioni implementate. Per i dettagli, consulta Implementazione di un allarme per rilevare automaticamente la deriva negli CloudFormation stack AWS.
- Cognito JSON Web Tokens (JWTs): la soluzione utilizza Amazon Cognito emesso da Amazon Cognito JWTs per l'autenticazione con gli endpoint dell'API REST. Abbiamo configurato la soluzione con una scadenza di cinque minuti per i token ID e i token di accesso. Quando un utente si disconnette, la sua capacità di generare nuovi token viene revocata (il token di aggiornamento viene revocato). Tuttavia, fino alla scadenza del token corrente, tutte le richieste all'endpoint API verranno autenticate con successo, poiché dispongono di un token valido. Esamina le considerazioni sulla sicurezza relative al tuo caso d'uso e modifica il periodo di validità del token.

Personalizzazione delle politiche del ciclo di vita:

Per le implementazioni di produzione, rivedi e modifica le politiche del ciclo di vita in base ai requisiti di conservazione. Vedi Impostazione della configurazione del ciclo di vita su un bucket nella Guida per l'utente di Amazon Simple Storage Service.

## Archiviazione e ciclo di vita dei file multimodali

Se hai abilitato le funzionalità di input multimodali (MultimodalEnabledimpostate su`Yes`) per il tuo caso d'uso, la soluzione crea un bucket Amazon S3 per archiviare i file caricati e una tabella DynamoDB per tenere traccia dei metadati dei file.

Politiche predefinite del ciclo di vita:

- File S3: eliminati automaticamente dopo 48 ore
- Metadati DynamoDB: i record scadono dopo 24 ore (cronologia delle conversazioni TTL)

Considerazioni sulla sicurezza:

- I file vengono partizionati in base all'ID del caso d'uso, all'ID utente, all'ID della conversazione e all'ID del messaggio e un file viene invece archiviato con un nome UUID. La mappatura dell'UUID ai nomi dei file è disponibile nella tabella dei metadati DynamoDB
- Gli utenti possono accedere solo ai file che hanno caricato all'interno delle proprie conversazioni
- La convalida del tipo di file viene eseguita utilizzando il rilevamento magico dei numeri
- Ti consigliamo di abilitare Amazon GuardDuty Malware Protection for S3 per scansionare i file caricati alla ricerca di contenuti dannosi

## Implementazione di un caso d'uso di testo autonomo

Segui le step-by-step istruzioni in questa sezione per configurare e distribuire la soluzione nel tuo account.

Tempo di implementazione: circa 10-30 minuti

1. Accedi alla Console di gestione AWS e seleziona il pulsante per avviare il CloudFront modello che desideri distribuire.

| | |
|---|---|
| BedrockChat.modello | Launch solution |
| SageMakerChat.modello | Launch solution |

2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra regione AWS, utilizza il selettore della regione nella barra di navigazione della console.

   Nota: questa soluzione utilizza Amazon Kendra e Amazon Bedrock, che attualmente non sono disponibili in tutte le regioni AWS. Se utilizzi queste funzionalità, devi avviare questa soluzione

in una regione AWS in cui questi servizi sono disponibili. Per la disponibilità più aggiornata per regione, consulta l'AWS Regional Services List.

3. Nella pagina Create stack *, verifica che l'URL del modello corretto sia nella casella di testo*Amazon S3 URL*e scegli *Avanti.
4. Nella pagina *Specificare i dettagli dello stack *, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni dei caratteri di denominazione, consulta IAM e STS Limits nella AWS Identity and Access Management User Guide.
5. In Parametri, esamina i parametri per questo modello di soluzione e modificali se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| | | |
|---|---|---|
| UseCaseUUID | *<_Requires input_>* | 36 caratteri di lunghezza UUIDv4 per identificare questo caso d'uso distribuito all'interno di un'applicazione. |
| UseCaseConfigRecordKey | *<_Requires input_>* | Chiave corrispondente al record contenente le configurazioni richieste dal provider di chat Lambda in fase di esecuzione. Il record nella tabella deve avere un attributo chiave che corrispon da a questo valore e un attributo config contenente la configurazione desiderat a. Questo record verrà compilato dalla piattaforma di distribuzione se in uso. Per le distribuzioni autonome di questo caso d'uso, è richiesta una voce creata manualmen te nella tabella definita in. UseCaseConfigTableName |
| UseCaseConfigTableName | *<_Requires input_>* | Lo stack leggerà la configura zione dalla tabella con |

| | | |
|---|---|---|
| | | questo nome come chiave UseCaseConfigRecordKey |

| | | |
|---|---|---|
| ExistingRestApiId | (Input opzionale) | ID API REST API Gateway esistente da utilizzare. Se non viene fornita, verrà creata una nuova API API Gateway REST. In genere viene fornita durante la distribuzione dalla dashboard di distribuzione.<br><br>Nota: l'utilizzo di APIs Existing può aiutare a ridurre la duplicazione delle risorse e semplificare la gestione APIs quando è necessario implementare più casi d'uso autonomi. Quando si fornisce una soluzione esistente APIs per un caso d'uso indipende nte, è responsabilità dell'uten te garantire che l'API sia configurata con le route richieste con i modelli previsti. È necessario configurare un percorso /details preconfig urato obbligatorio (recupera i dettagli dei casi d'uso durante la chat) e, facoltativamente, un percorso /feedback (se impostato per consentire FeedbackEnabledla raccolta di feedback per le risposte `Yes` alle chat LLM). Inoltre, e deve anche essere ExistingA piRootResourceIdfornito ExistingCognitoUserPoolId. ExistingCognitoGroupPolicyT ableName |

| | | |
|---|---|---|
| ExistingApiRootResourceId | (Input opzionale) | ID risorsa principale dell'API REST di API Gateway esistente da utilizzare. Il Root Resource ID dell'API REST può essere ottenuto dalla console AWS selezionando la risorsa root (/) nella sezione «Risorse» dell'API. L'ID della risorsa verrà quindi visualizz ato nel pannello dei dettagli della risorsa. In alternativa, puoi eseguire una chiamata API di descrizione sull'API REST per trovare il Root Resource ID. |
| FeedbackEnabled | No | Se impostato su No, lo stack di use case distribuito non avrà accesso alla funzionalità di feedback. |
| ExistingModelInfoTableName | (Input opzionale) | Nome della tabella DynamoDB per la tabella che contiene informazioni sul modello e valori predefini ti. Utilizzato dalla piattafor ma di distribuzione. Se omessa, verrà creata una nuova tabella per contenere le impostazioni predefinite del modello. |

| | | |
|---|---|---|
| DefaultUserEmail | `placeholder@example.com` | E-mail dell'utente predefinito per questo caso d'uso. Viene creato un utente Amazon Cognito per questa e-mail per accedere allo use case. Se non vengono forniti, il Gruppo e l'Utente di Cognito non verranno creati. Puoi anche utilizzare `placeholder@example.com` per creare il gruppo ma non l'utente. Per informazioni sulla configurazione del pool di utenti, fare riferimento alla sezione Configurazione manuale del pool di utenti. |
| ExistingCognitoUserPoolId | (Input opzionale) | UserPoolIddi un pool di utenti Amazon Cognito esistente con cui verrà autenticato questo caso d'uso. In genere viene fornito durante la distribuzione dalla dashboard di Deployment, ma può essere omesso quando si distribuisce questo stack di use case in modo autonomo. |
| CognitoDomainPrefix | (Input opzionale) | Inserisci un valore se desideri fornire un dominio per il client del pool di utenti di Cognito. Se non fornisci un valore, l'implementazione ne genererà uno. |

| | | |
|---|---|---|
| ExistingCognitoUserPoolClient | (Input opzionale) | Fornisci un client del pool di utenti (App Client) per utilizzarne uno esistente. Se non fornisci un User Pool Client, ne verrà creato uno nuovo. Questo parametro può essere fornito solo se viene fornito un ID del pool di utenti esistente. |
| ExistingCognitoGroupPolicyTableName | (Input opzionale) | Nome della tabella DynamoDB contenente le politiche dei gruppi di utenti. Viene utilizzato dall'autorizzatore personalizzato sull'API del caso d'uso. In genere, è possibile fornire un input durante la distribuzione dalla piattaforma di distribuzione, ma può essere omesso quando si distribuisce questo stack di casi d'uso in modo autonomo. |
| RAGEnabled | `true` | Se impostato su true, lo stack di use case distribuito utilizza l'indice Amazon Kendra fornito creato per fornire la funzionalità RAG. Se impostato su`false`, l'utente interagisce direttamente con il LLM. |

| | | |
|---|---|---|
| KnowledgeBaseType | Bedrock | Tipo di knowledge base da utilizzare per RAG. Imposta solo se lo RAGEnabled è`true`. Può essere Bedrock o Kendra.<br><br>Nota: Rilevante solo se RAGEnabledè vero. |
| ExistingKendraIndexId | (Input opzionale) | ID indice di un indice Kendra esistente da utilizzare per il caso d'uso. Se non ne viene fornito nessuno ed KnowledgeBaseTypeè Kendra, verrà creato un nuovo indice per te.<br><br>Nota: rilevante solo se RAGEnabledè `true` ed KnowledgeBaseTypeè. `Kendra` |
| NewKendraIndexName | (Inserimento opzionale) | Nome per il nuovo indice Kendra da creare per questo caso d'uso. Si applica solo se non ExistingKendraInde xIdviene fornito.<br><br>Nota: rilevante solo se RAGEnabledè vero ed KnowledgeBaseTypeè Kendra. |

| | | |
|---|---|---|
| NewKendraQueryCapacityUnits | 0 | Unità di capacità di query aggiuntive per il nuovo indice Amazon Kendra da creare per questo caso d'uso. Si applica solo se non ExistingKendraIndexIdviene fornito, vedi. CapacityUnitsConfiguration<br><br>Nota: rilevante solo se RAGEnabledè `true` ed Knowledge BaseTypeè`Kendra`. |
| NewKendraStorageCapacityUnits | `0` | Unità di capacità di storage aggiuntive per il nuovo indice Amazon Kendra da creare per questo caso d'uso. Si applica solo se non ExistingKendraIndexIdviene fornito, vedi. CapacityUnitsConfiguration<br><br>Nota: rilevante solo se RAGEnabledè `true` ed Knowledge BaseTypeè`Kendra`. |

| | | |
|---|---|---|
| NewKendraIndexEdition | (Inserimento opzionale) | L'edizione di Amazon Kendra da utilizzare per il nuovo indice Amazon Kendra da creare per questo caso d'uso. Si applica solo se non ExistingKendraIndexIdviene fornito, vedi Amazon Kendra Editions.<br><br>Nota: rilevante solo se RAGEnabledè `true` ed KnowledgeBaseTypeè. `Kendra` |
| BedrockKnowledgeBaseId | (Inserimento opzionale) | ID della knowledge base da utilizzare in un caso d'uso RAG. Non può essere fornito se ExistingKendraIndexIdo NewKendraIndexNameviene fornito.<br><br>Nota: rilevante solo se RAGEnabledè `true` ed Knowledge BaseTypeè`Bedrock`. |
| VpcEnabled | `No` | Se le risorse degli stack devono essere distribuite all'interno di un VPC. |
| CreateNewVpc | `No` | Seleziona`Yes`, se desideri che la soluzione crei un nuovo VPC per te e venga utilizzata per questo caso d'uso.<br><br>Nota: rilevante solo se lo VpcEnabledè`Yes`. |

| | | |
|---|---|---|
| IPAMPoolId | (Inserimento opzionale) | Se desideri assegnare l'intervallo CIDR utilizzando Amazon VPC IP Address Manager, fornisci l'ID del pool IPAM da utilizzare.<br><br>Nota: rilevante solo se VpcEnabledè ed è. `Yes` CreateNewVpc`No` |
| ExistingVpcId | (Inserimento opzionale) | ID VPC di un VPC esistente da utilizzare per lo use case.<br><br>Nota: rilevante solo se VpcEnabledè `Yes` ed CreateNewVpcè. `No` |
| ExistingPrivateSubnetIds | (Inserimento opzionale) | Elenco separato da virgole IDs di sottoreti private esistenti da utilizzare per distribuire la funzione Lambda.<br><br>Nota: rilevante solo se è ed è. VpcEnabled**`Yes`**CreateNew Vpc`No` |
| ExistingSecurityGroupIds | (Inserimento opzionale) | Elenco separato da virgole dei gruppi di sicurezza del VPC esistente da utilizzare per configurare le funzioni Lambda.<br><br>Nota: rilevante solo se VpcEnabledè ed è`Yes`. CreateNewVpc`No` |

| | | |
|---|---|---|
| VpcAzs | (Inserimento opzionale) | Elenco separato da virgole AZs in cui vengono create le sottoreti di VPCs<br><br>Nota: rilevante solo se VpcEnabledè `Yes` ed CreateNewVpcè. `No` |
| UseInferenceProfile | `No` | Se il modello configurato è Bedrock, puoi indicare se stai utilizzando Bedrock Inference Profile. Ciò garantirà che le politiche IAM richieste vengano configurate durante la distribuzione dello stack. Per maggiori dettagli, consulta il seguente file - region-inference.html https://docs.aws.amazon.com/bedrock/ latest/userguide/cross |
| Implementa UI | Sì | Seleziona l'opzione per distribuire l'interfaccia utente frontend per questa distribuzione. Selezionando No, verrà creata solo l'infrastruttura per ospitare l' APIselaborazione, l'autenticazione e il APIs backend. |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella per confermare che il modello creerà risorse AWS Identity and Access Management (IAM).
9. Seleziona Create (Crea) per implementare lo stack.

Puoi visualizzare lo stato dello stack nella CloudFormation console AWS nella colonna Status. Dovresti ricevere lo stato CREATE_COMPLETE in circa 10-30 minuti.

## Implementazione di un caso d'uso autonomo di Bedrock Agent

Segui le step-by-step istruzioni in questa sezione per configurare e distribuire la soluzione nel tuo account.

Tempo di implementazione: circa 10-30 minuti

1. Accedi alla Console di gestione AWS e seleziona il pulsante per avviare il CloudFront modello.

| | |
|---|---|
| BedrockAgent.modello | Launch solution |

2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra regione AWS, utilizza il selettore della regione nella barra di navigazione della console.

> **Note**
>
> Questa soluzione utilizza Amazon Bedrock, che attualmente non è disponibile in tutte le regioni AWS. Se utilizzi queste funzionalità, devi avviare questa soluzione in una regione AWS in cui questi servizi sono disponibili. Per la disponibilità più aggiornata per regione, consulta l'AWS Regional Services List.

3. Nella pagina Create stack, verifica che l'URL del modello corretto sia nella casella di testo URL Amazon S3 e scegli Avanti.
4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni dei caratteri di denominazione, consulta {https---docs-aws-amazon-com- https---docs-aws-amazon-com -iam-Latest- UserGuide -reference-iam-limits-html} [quote IAM e AWS STS] nella AWS Identity and Access Management User Guide.
5. In Parametri, esamina i parametri per questo modello di soluzione e modificali se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Voce predefinita | Description |
| --- | --- | --- |
| UseCaseUUID | *<_Requires input_>* | 36 caratteri di lunghezza UUIDv4 per identificare questo caso d'uso distribuito all'interno di un'applicazione. |
| UseCaseConfigRecordKey | *<Requires input>* | Chiave corrispondente al record che contiene le configurazioni richieste dalla funzione Lambda del provider di chat in fase di esecuzione.<br><br>Il record nella tabella deve avere un attributo chiave che corrisponda a questo valore e un attributo config contenente la configurazione desiderata.<br><br>Questo record verrà compilato dalla piattaforma di distribuzione se è in uso. Per le distribuzioni autonome di questo caso d'uso, è richiesta una voce creata manualmen te nella tabella definita in. UseCaseConfigTableName |
| UseCaseConfigTableName | *<Requires input>`* | Lo stack leggerà la configura zione del caso d'uso dalla tabella fornita qui e utilizzan do la chiave di registraz ione definita in. UseCaseCo nfigRecordKey |

| Parametro | Voce predefinita | Description |
| --- | --- | --- |
| DefaultUserEmail | `placeholder@example.com` | E-mail dell'utente predefinito per questo caso d'uso. La soluzione crea un utente Amazon Cognito per questa e-mail per accedere allo use case. |

| Parametro | Voce predefinita | Description |
| --- | --- | --- |
| ExistingRestApiId | (Inserimento opzionale) | ID API REST API Gateway esistente da utilizzare. Se non viene fornita, verrà creata una nuova API API Gateway REST. In genere viene fornita durante la distribuzione dalla dashboard di distribuzione.<br><br>Nota: l'utilizzo di APIs Existing può aiutare a ridurre la duplicazione delle risorse e semplificare la gestione APIs quando è necessario implementare più casi d'uso autonomi. Quando si fornisce una soluzione esistente APIs per un caso d'uso indipende nte, è responsabilità dell'uten te garantire che l'API sia configurata con le route richieste con i modelli previsti. È necessario configurare un percorso /details preconfig urato obbligatorio (recupera i dettagli dei casi d'uso durante la chat) e, facoltativamente, un percorso /feedback (se impostato per consentire FeedbackEnabledla raccolta di feedback per le risposte `Yes` alle chat LLM). Inoltre, e deve anche essere ExistingA piRootResourceIdfornito ExistingCognitoUserPoolId. |

| Parametro | Voce predefinita | Description |
| --- | --- | --- |
| | | ExistingCognitoGroupPolicyTableName |
| ExistingApiRootResourceId | (Input opzionale) | ID risorsa principale dell'API REST di API Gateway esistente da utilizzare. Il Root Resource ID dell'API REST può essere ottenuto dalla console AWS selezionando la risorsa root (/) nella sezione «Risorse» dell'API. L'ID risorsa verrà quindi visualizzato nel pannello dei dettagli della risorsa. In alternativa, puoi eseguire una chiamata API description sulla tua API REST per trovare il Root Resource ID. |
| FeedbackEnabled | No | Se impostato su No, lo stack di use case distribuito non avrà accesso alla funzionalità di feedback. |
| CognitoDomainPrefix | (Input opzionale) | Inserisci un valore se desideri fornire un dominio per il client del pool di utenti di Amazon Cognito. Se non fornisci un valore, la soluzione ne genera uno. |

| Parametro | Voce predefinita | Description |
| --- | --- | --- |
| ExistingCognitoUserPoolId | (Input opzionale) | UserPoolIddi un pool di utenti Amazon Cognito esistente con cui desideri autenticare questo caso d'uso. NOTA: in genere fornisci questo ID durante la distribuzione dalla dashboard di Deployment, ma puoi ometterlo quando distribuisci questo stack di use case standalone. |
| ExistingCognitoUserPoolClient | (Input opzionale) | Fornisci un client del pool di utenti (client dell'app) per utilizzarne uno esistente. Se non fornisci un client per il pool di utenti, la soluzione ne crea uno. Puoi fornire questo parametro solo se hai fornito un ExistingCognitoUserPoolId. |
| ExistingCognitoGroupPolicyTableName | (Input opzionale) | Nome della tabella DynamoDB contenent e le politiche dei gruppi di utenti. Viene utilizzato dall'autorizzatore personalizzato sull'API del caso d'uso. NOTA: in genere si fornisce questo nome durante la distribuzione dalla dashboard di Deploymen t, ma è possibile ometterlo quando si distribuisce questo stack di use case standalone. |

| Parametro | Voce predefinita | Description |
| --- | --- | --- |
| VpcEnabled | No | Se le risorse degli stack devono essere distribuite all'interno di un VPC. |
| CreateNewVpc | No | Seleziona Yes se desideri che la soluzione crei un nuovo VPC per te e lo utilizzi per questo caso d'uso. NOTA: questo parametro è rilevante solo se lo VpcEnabledèYes. |
| IPAMPoolId | (Input opzionale) | Se desideri assegnare l'intervallo CIDR utilizzando IPAM, fornisci l'ID del pool IPAM da utilizzare. NOTA: questo parametro è rilevante solo se VpcEnabledè ed è. Yes CreateNewVpcNo |
| ExistingVpcId | (Input opzionale) | ID VPC di un VPC esistente da utilizzare per lo use case. NOTA: questo parametro è rilevante solo se VpcEnable dè Yes ed CreateNewVpcè. No |
| ExistingPrivateSubnetIds | (Input opzionale) | Elenco separato da virgole IDs di sottoreti private esistenti da utilizzare per distribuire la funzione Lambda. NOTA: questo parametro è rilevante solo se è ed è. VpcEnable d**Yes**CreateNewVpcNo |

| Parametro | Voce predefinita | Description |
| --- | --- | --- |
| ExistingSecurityGroupIds | (Input opzionale) | Elenco separato da virgole dei gruppi di sicurezza del VPC esistente da utilizzare per configurare le funzioni Lambda. NOTA: questo parametro è rilevante solo se è ed è. VpcEnable d**`Yes`**CreateNewVpcNo |
| VpcAzs | (Input opzionale) | Elenco separato da virgole AZs in cui vengono create le sottoreti di VPCs<br><br>Nota: rilevante solo se VpcEnabledè `Yes` ed CreateNewVpcè. `No` |
| BedrockAgentId | *`<Requires input>`* | L'ID dell'agente Amazon Bedrock da utilizzare. |
| BedrockAgentAliasId | *`<Requires input>`* | L'ID alias dell'agente Amazon Bedrock da utilizzare. |
| Implementa l'interfaccia utente | `Yes` | Seleziona l'opzione per implementare l'interfaccia utente di chat frontend per questa distribuzione. La selezione No comporta la creazione dell'infrastruttura per ospitare APIs, l'autenti cazione per l'elaborazione e il APIs backend senza l'interfa ccia utente della chat. |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).

8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella per confermare che il modello creerà risorse IAM.
9. Seleziona Create (Crea) per implementare lo stack.

Puoi visualizzare lo stato dello stack nella CloudFormation console AWS nella colonna Status. Dovresti ricevere lo stato CREATE_COMPLETE in circa 10-30 minuti.

# Fornire una configurazione di chat DynamoDB

Quando si implementa un caso d'uso, UseCaseConfigRecordKeysono CloudFormation parametri obbligatori che normalmente UseCaseConfigTableNamevengono compilati dal pannello di controllo Deployment. Lo stack dei dashboard di distribuzione gestisce la creazione e la configurazione di questa tabella, mentre le chiamate all'API di distribuzione attivano la compilazione dei parametri.

Quando si esegue una distribuzione autonoma, è necessario effettuare le seguenti operazioni:

1. Crea una tabella DynamoDB con una chiave hash o chiave.
2. Crea un record nella tabella contenente la configurazione per lo use case come record del formato: `{key: some_use_case_key, config: {your_configuration}.`
3. Passa i parametri scelti UseCaseConfigTableNamee UseCaseConfigRecordKey(`some_use_case_key`in questo esempio) allo stack di use case durante la distribuzione.

Per creare una configurazione adatta per una distribuzione autonoma, puoi creare uno use case richiesto dalla dashboard di Deployment e copiare il record dalla tabella di configurazione. Altrimenti, puoi creare la tua configurazione sulla base del seguente esempio per una distribuzione di Bedrock:

```
{
 "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
 }
}
```

# Monitora la soluzione con Service Catalog AppRegistry

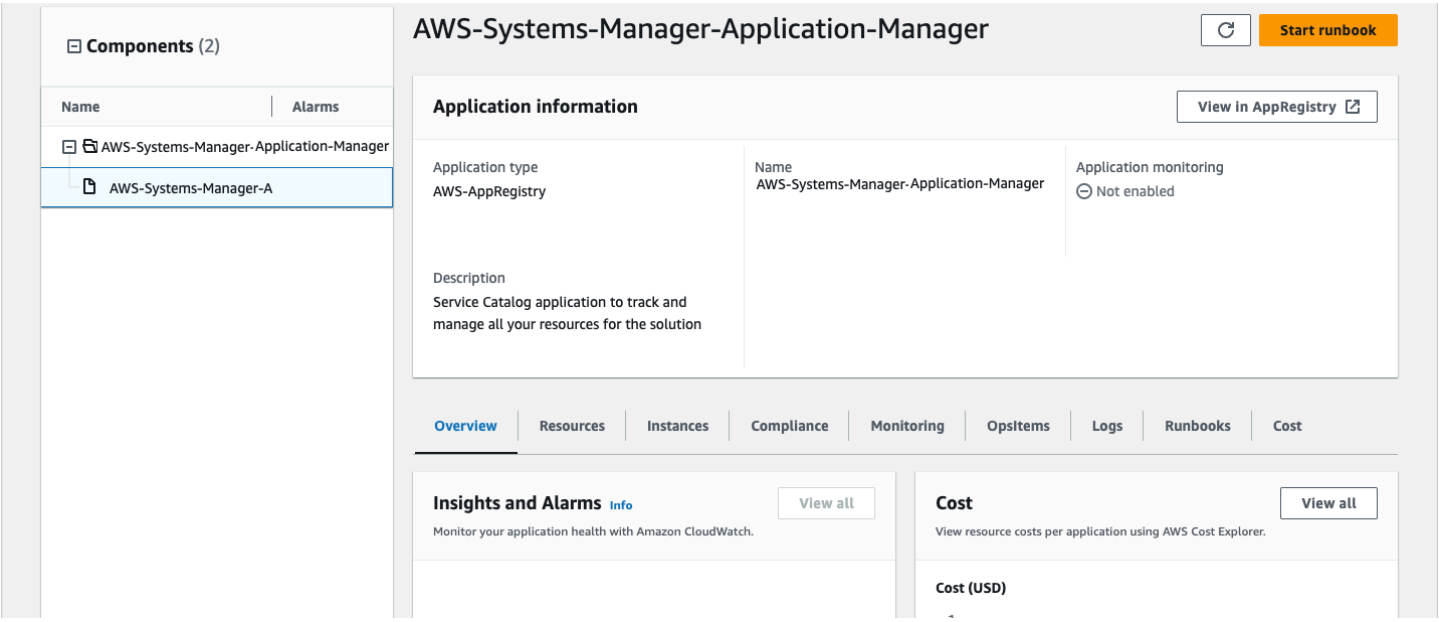
La soluzione include una AppRegistry risorsa Service Catalog per registrare il CloudFormation modello e le risorse sottostanti come applicazione sia in Service Catalog AppRegistry che in Systems Manager Application Manager.

Systems Manager Application Manager offre una visione a livello di applicazione di questa soluzione e delle relative risorse in modo da poter:

- Monitora le risorse, i costi delle risorse distribuite su stack e account AWS e i log associati a questa soluzione da una posizione centrale.
- Visualizza i dati operativi per le risorse di questa soluzione nel contesto di un'applicazione. Ad esempio, lo stato dell'implementazione, gli CloudWatch allarmi, le configurazioni delle risorse e i problemi operativi.

La figura seguente mostra un esempio di visualizzazione delle applicazioni per lo stack di soluzioni in Application Manager.

Illustra lo stack di soluzioni in Application Manager



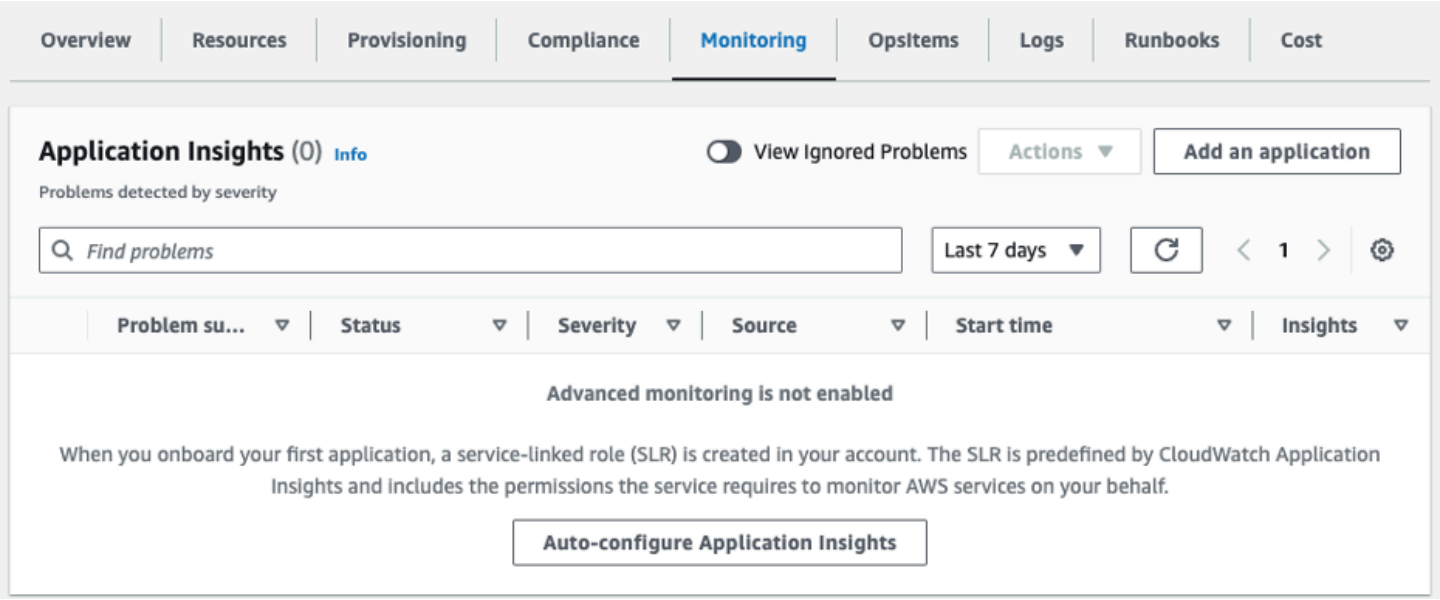

# Attiva Application Insights CloudWatch

1. Accedere alla console Systems Manager.

2. Nel riquadro di navigazione, scegli Application Manager.
3. In Applicazioni, cerca il nome dell'applicazione per questa soluzione e selezionalo.

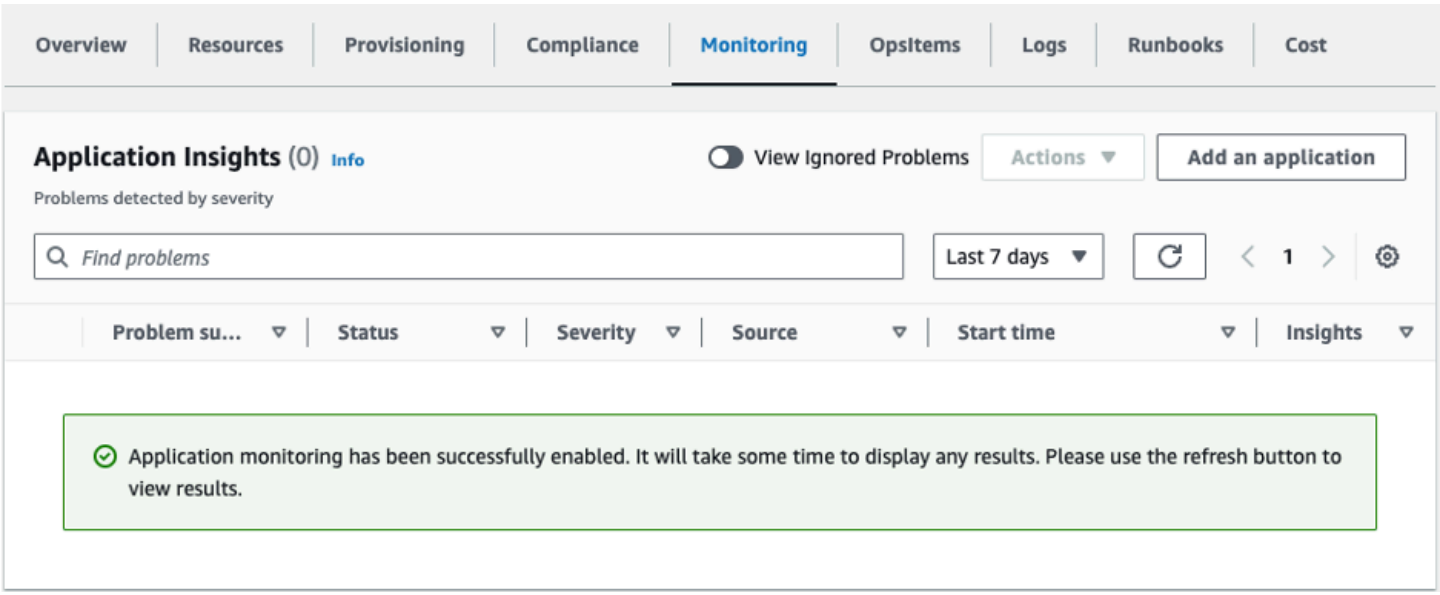
   Il nome dell'applicazione avrà il registro delle app nella colonna Origine dell'applicazione e avrà una combinazione del nome della soluzione, della regione, dell'ID dell'account o del nome dello stack.
4. Nell'albero dei componenti, scegliete lo stack di applicazioni che desiderate attivare.
5. Nella scheda Monitoraggio, in Application Insights, seleziona Configura automaticamente Application Insights.

Dashboard di Application Insights che mostra l'assenza di problemi rilevati e l'opzione di configurazione automatica.




Il monitoraggio delle applicazioni è ora attivato e viene visualizzata la seguente casella di stato:

La dashboard di Application Insights mostra il messaggio di avvenuta attivazione del monitoraggio.



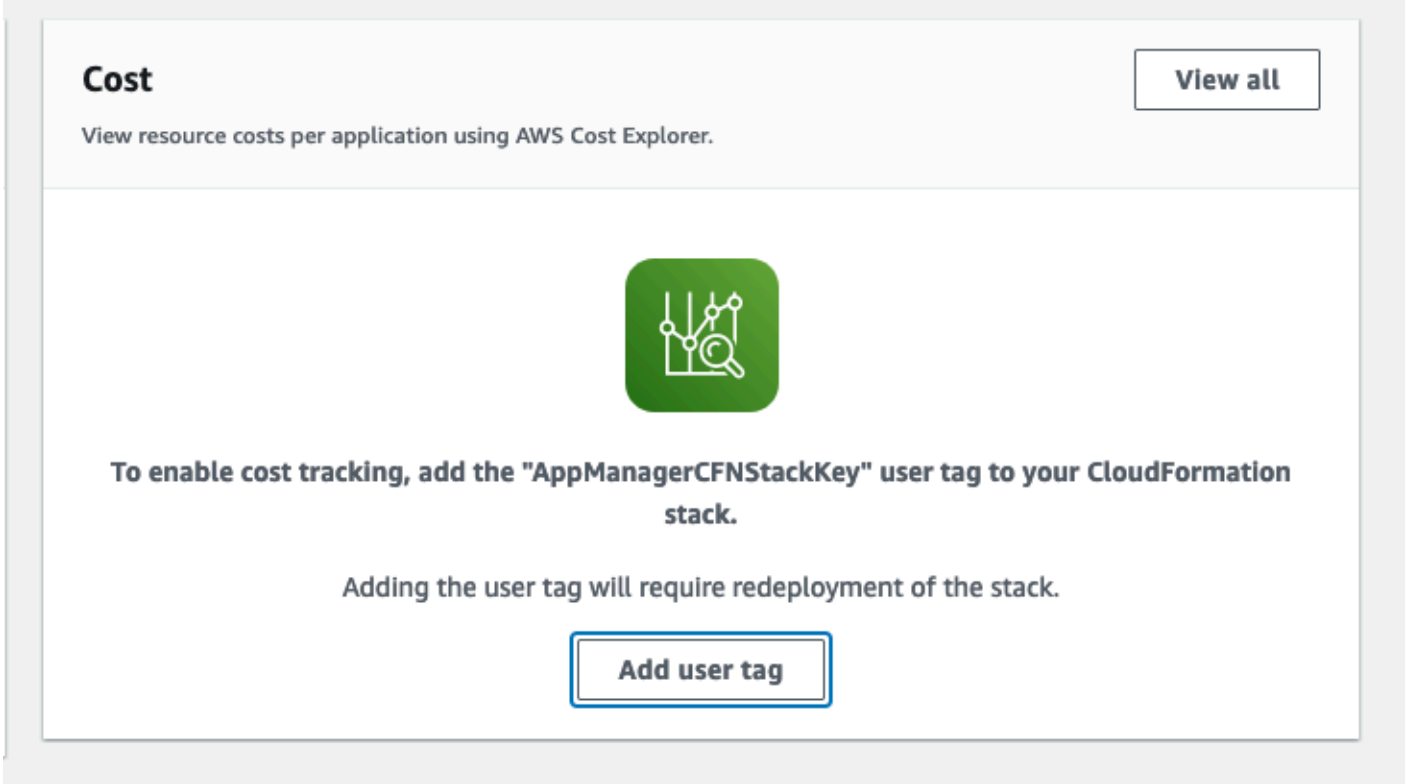

# Conferma i cartellini dei costi associati alla soluzione

Dopo aver attivato i tag di allocazione dei costi associati alla soluzione, è necessario confermare i tag di allocazione dei costi per visualizzare i costi di questa soluzione. Per confermare i tag di allocazione dei costi:

1. Accedere alla console Systems Manager.
2. Nel riquadro di navigazione, scegli Application Manager.
3. In Applicazioni, scegli il nome dell'applicazione per questa soluzione e selezionala.

   Il nome dell'applicazione avrà il registro delle app nella colonna Origine dell'applicazione e avrà una combinazione del nome della soluzione, della regione, dell'ID dell'account o del nome dello stack.
4. Nella scheda Panoramica, in Costo, seleziona Aggiungi tag utente.

   Schermata che mostra la schermata Application Cost Add User Tag




5. Nella pagina Aggiungi tag utente, inserisci`confirm`, quindi seleziona Aggiungi tag utente.

Il completamento del processo di attivazione può richiedere fino a 24 ore e la visualizzazione dei dati del tag.

## Attiva i tag di allocazione dei costi associati alla soluzione

Dopo aver attivato Cost Explorer, è necessario attivare i tag di allocazione dei costi associati a questa soluzione per visualizzare i costi di questa soluzione. I tag di allocazione dei costi possono essere attivati solo dall'account di gestione dell'organizzazione. Per attivare i tag di allocazione dei costi:

1. Accedi alla console AWS Billing and Cost Management and Cost Management.
2. Nel riquadro di navigazione, seleziona Cost Allocation Tags.
3. Nella pagina Tag di allocazione dei costi, filtra per il tag AppManager CFNStack Chiave, quindi seleziona il tag dai risultati visualizzati.
4. Selezionare Attiva.

# AWS Cost Explorer

È possibile visualizzare la panoramica dei costi associati all'applicazione e ai componenti dell'applicazione all'interno della console di Application Manager tramite l'integrazione con AWS Cost Explorer, che deve essere prima attivato. Cost Explorer ti aiuta a gestire i costi fornendo una panoramica dei costi e dell'utilizzo delle risorse AWS nel tempo. Per attivare Cost Explorer per la soluzione:

1. Accedi alla console AWS Cost Management.
2. Nel riquadro di navigazione, seleziona Cost Explorer per visualizzare i costi e l'utilizzo della soluzione nel tempo.

# Aggiorna la soluzione

Se hai già distribuito la soluzione, segui questa procedura per aggiornare lo CloudFormation stack della soluzione e ottenere le funzionalità e i miglioramenti più recenti. Il processo di aggiornamento è suddiviso in tre parti:

- Fase 1: Aggiornamento del pannello di distribuzione
- Fase 2: Migrazione delle configurazioni dei casi d'uso
- Fase 3: Aggiornamento dei casi d'uso

> **Note**
>
> 1. Nella versione 2.0.0, l'integrazione con Anthropic e Hugging Face era obsoleta a favore di Amazon Bedrock e Amazon AI. SageMaker Puoi implementare modelli disponibili tramite Hugging Face tramite. SageMaker JumpStart Per maggiori dettagli, consulta Use Hugging Face with Amazon SageMaker AI.
> 2. Assicurati di testare il processo di aggiornamento in un ambiente non di produzione prima di eseguire questi passaggi.

## Fase 1: Aggiornamento del pannello di distribuzione

1. Accedi alla CloudFormation console, seleziona lo CloudFormation stack esistente e seleziona Aggiorna.
2. Seleziona Sostituisci modello corrente.
3. In Specificare il modello:
   a. Seleziona l'URL di Amazon S3.
   b. Copia il link del CloudFormation modello più recente.
   c. Incolla il link nella casella dell'URL di Amazon S3.
   d. Verifica che l'URL del modello corretto sia visualizzato nella casella di testo URL di Amazon S3 e scegli Avanti. Scegliere Next (Successivo) di nuovo.
4. In Parametri, esamina i parametri del modello e modificali se necessario. Per i dettagli sui parametri, consulta Fase 1: Avvio dello stack di dashboard di distribuzione.

5. Scegli Next (Successivo).
6. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
7. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella per confermare che il modello creerà risorse IAM.
8. Scegli Visualizza set di modifiche e verifica le modifiche.
9. Scegli Aggiorna stack per distribuire lo stack.

Puoi visualizzare lo stato dello stack nella CloudFormation console AWS nella colonna Status. Dovresti ricevere lo stato UPDATE_COMPLETE in circa 10 minuti.

Se la versione esistente della Soluzione era precedente alla v2.0.0, l'aggiornamento creerà uno stack di interfaccia utente Web (che sostituisce l'`amplify-ui`implementazione della schermata di accesso con un'interfaccia utente ospitata da Cognito) e un nuovo CloudFront URL, che può essere ottenuto dalla sezione Output della CloudFormation console una volta che lo stato dello stack è UPDATE_COMPLETE.

> **Note**
>
> I casi d'uso esistenti creati utilizzando versioni precedenti alla v2.0.0 NON verranno visualizzati finché non completerai i passaggi descritti di seguito.

## Fase 2: Migrazione delle configurazioni dei casi d'uso (solo aggiornamenti da versioni precedenti alla 2.0.0)

Lo schema per lo storage e la configurazione dei casi d'uso del servizio AWS per l'archiviazione sono stati modificati nella versione 2.0.0. Segui i passaggi descritti nella GAAB v2 Migration User Guide utilizzando lo script gaab_v2_migration.py. Dopo aver eseguito lo script, puoi accedere alla dashboard di Deployment per visualizzare i casi d'uso distribuiti.

> **Note**
>
> È necessario seguire i passaggi seguenti per completare la migrazione dei casi d'uso.

# Fase 3: Aggiornamento dei casi d'uso

È possibile modificare i casi d'uso implementati con le nuove funzionalità disponibili nelle ultime versioni di GAAB. Vedi Utilizzare la soluzione per informazioni su come utilizzare le funzionalità di questa soluzione.

Per aggiornare i casi d'uso alla versione più recente, è necessario completare la procedura `Modifica`relativa ai casi d'uso nella dashboard di distribuzione (anche se è possibile che non vengano apportate modifiche). Questa azione attiva un aggiornamento CloudFormation dello stack con l'ultima versione del modello.

> **Note**
>
> I casi d'uso creati con le versioni 1.x o 2.x della soluzione potrebbero non funzionare con le versioni successive. Pertanto, consigliamo di clonare i casi d'uso esistenti creati con versioni precedenti alla v3.0.0 tramite la dashboard di distribuzione. Quindi, esegui gradualmente la migrazione e la sostituzione con nuovi use case creati utilizzando la versione 3.0.0 o successiva.

# Risoluzione dei problemi

Questa sezione fornisce istruzioni per la risoluzione dei problemi relativi alla distribuzione e all'utilizzo della soluzione.

Se queste istruzioni non risolvono il problema, Contact Support fornisce le istruzioni per aprire una richiesta di assistenza per questa soluzione.

## Problema: l'implementazione di una configurazione abilitata per VPC, con Create a VPC for me, non riesce

Lo stack di dashboard Deployment o lo stack di use case non vengono implementati perché non CloudFormation è stato possibile fornire risorse di rete VPC.

### Risoluzione

Controlla i limiti di quota per VPCs ed Elastic IPs nel tuo account. I limiti predefiniti sono 5 ciascuno per Elastic IPs e VPCs per account AWS, per regione AWS.

> **Note**
>
> Quando la soluzione crea un VPC, una singola implementazione abilitata per VPC (Deployment Dashboard o Use Case) è una distribuzione 2-AZ con 1 sottorete pubblica e 1 privata in ogni AZ, ogni sottorete pubblica implementa 1 gateway NAT. Con 2 gateway NAT, l'implementazione utilizza 2 indirizzi IP pubblici rispetto al limite di quota.

Alcuni limiti da tenere a mente (per account, per regione):

- Numero di VPCs - 5
- Numero di indirizzi IP pubblici: 5
- Numero di endpoint VPC gateway: 20
- Numero di endpoint VPC di interfaccia: 20

# Problema: lo stack di use case non può essere eliminato CloudFormation dopo l'eliminazione dello stack del dashboard di Deployment

Se lo stack del dashboard di Deployment viene eliminato CloudFormation prima che tutti gli stack di use case vengano eliminati, gli use case possono finire in uno stato bloccato (inutilizzabile). Ciò è dovuto al fatto che un ruolo IAM creato dallo stack di dashboard Deployment non esiste più e impedisce le modifiche allo stack di use case.

## Risoluzione

> **Warning**
>
> Assicurati di ripulire tutti i ruoli creati manualmente subito dopo l'uso. Si tratta di autorizzazioni elevate che gli utenti potrebbero sfruttare per l'elevazione dei ruoli.

Ricrea il ruolo IAM eliminato per consentire l'eliminazione degli stack: CloudFormation

1. Apri la CloudFormation console e determina il ruolo associato allo stack bloccato.
   a. Il ruolo ARN può essere trovato nella sezione delle informazioni sullo stack denominata ruolo IAM.
   b. Il nome del ruolo è quello che segue dopo: role/ nel ruolo IAM ARN (ad esempio, arn:aws:iam: :role/) <account-id><role-name>
2. Crea un nuovo ruolo in IAM con lo stesso nome del ruolo eliminato.
   a. Seleziona il servizio AWS come entità affidabile e CloudFormationselezionalo dal menu a discesa.
   b. Aggiungi le autorizzazioni necessarie. Se non sei sicuro delle autorizzazioni richieste, puoi utilizzare la policy gestita AdministratorAccessda AWS.
   c. Inserisci il nome del ruolo esattamente come ottenuto nella Fase 1.
3. Torna alla CloudFormation console ed elimina gli stack bloccati.
4. Una volta che tutti gli stack bloccati sono stati eliminati con successo, torna a IAM ed elimina tutti i ruoli creati nello Step 2.

# Problema: l'interfaccia utente dei casi d'uso non riflette le modifiche nelle impostazioni

Quando i casi d'uso vengono aggiornati, l'interfaccia utente viene distribuita su. CloudFront Tuttavia, poiché CloudFront memorizza nella cache le distribuzioni e il file di configurazione che determina il modo in cui alcune impostazioni vengono mostrate all'utente, queste modifiche potrebbero non riflettersi immediatamente.

## Risoluzione

La CloudFront distribuzione può essere invalidata per forzare la propagazione della nuova configurazione agli utenti del frontend.

1. Apri la CloudFormation console e determina la CloudFront distribuzione associata allo stack di use case.
   a. Lo stack di casi d'uso dovrebbe iniziare con lo stesso nome utilizzato durante la distribuzione dello use case.
   b. Individua lo stack annidato corrispondente all'interfaccia utente. Il nome dello stack annidato deve iniziare con S3 StackS3. WebApp UINested UINested StackResource
   c. Nella scheda Risorse, individua il tipo di risorsa AWS::CloudFront::Distribution, quindi seleziona l'ID fisico. Questo aprirà la distribuzione nella CloudFront console.
2. Vai alla scheda Invalidazioni, quindi scegli Crea invalidazione e inserisci un percorso di /*. Questo invaliderà tutti i percorsi.
3. Nel tuo browser, elimina tutti i cookie e i file memorizzati nella cache relativi al caso d'uso.

# Contattare AWS Support

Se disponi di AWS Business Support+, AWS Enterprise Support o Unified Operations, puoi utilizzare l'AWS Support Center per ottenere l'assistenza di esperti su questa soluzione. Le istruzioni per eseguire tali operazioni sono fornite nelle sezioni seguenti.

## Crea un caso

1. Accedi al Support Center.
2. Scegli Crea caso.

## Come possiamo aiutarti?

1. Scegli Tecnico.
2. Per Assistenza, seleziona Soluzioni.
3. Per Categoria, seleziona Altre soluzioni.
4. Per Severità, seleziona l'opzione più adatta al tuo caso d'uso.
5. Quando si inseriscono i campi Servizio, Categoria e Severità, l'interfaccia compila i collegamenti alle domande più comuni per la risoluzione dei problemi. Se non riesci a risolvere la tua domanda con questi link, scegli Passaggio successivo: Informazioni aggiuntive.

## Informazioni aggiuntive

1. In Oggetto, inserisci il testo che riassume la domanda o il problema.
2. Per la descrizione, descrivi il problema in dettaglio, includendo il nome di questa soluzione: Generative AI Application Builder on AWS.
3. Scegli Allega file.
4. Allega le informazioni di cui AWS Support ha bisogno per elaborare la richiesta.

## Aiutaci a risolvere il tuo caso più velocemente

1. Inserisci le informazioni richieste.
2. Scegli Passaggio successivo: risolvi ora o contattaci.

## Risolvi subito o contattaci

1. Rivedi le soluzioni Solve now.
2. Se non riesci a risolvere il problema con queste soluzioni, scegli Contattaci, inserisci le informazioni richieste e scegli Invia.

# Disinstalla la soluzione

> **Note**
>
> Le distribuzioni create tramite la dashboard di Deployment non sono destinate a essere gestite all'esterno della soluzione. Assicurati di eliminare e ripulire tutte le distribuzioni dalla dashboard di Deployment, prima di eliminare lo stack in esso contenuto. CloudFormation

Puoi disinstallare la soluzione Generative AI Application Builder on AWS dalla Console di gestione AWS o utilizzando l'interfaccia a riga di comando AWS. È necessario eliminare manualmente i bucket Amazon S3, gli indici Amazon Kendra o i log creati da questa soluzione. CloudWatch Le soluzioni AWS non eliminano automaticamente i bucket Amazon S3, gli indici Amazon Kendra o i CloudWatch log nel caso in cui siano archiviati dati da conservare.

## Utilizzando la Console di gestione AWS

1. Accedi alla CloudFormation console AWS.
2. Nella pagina Stacks, seleziona lo stack di installazione di questa soluzione.
3. Scegli Elimina.

## Utilizzo dell'interfaccia a riga di comando AWS

Determina se l'AWS Command Line Interface (AWS CLI) è disponibile nel tuo ambiente. Per istruzioni di installazione, consulta What Is the Command Line Interface di AWS nella AWS CLI User Guide. Dopo aver verificato che la CLI di AWS è disponibile, esegui il comando seguente.

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

## Procedura di disinstallazione manuale

### Eliminazione dei bucket Amazon S3

Questa soluzione è configurata per conservare il bucket Amazon S3 creato dalla soluzione se decidi di eliminare lo stack CloudFormation AWS per prevenire la perdita accidentale di dati. Dopo aver

disinstallato la soluzione, puoi eliminare manualmente questo bucket Amazon S3 se non hai bisogno di conservare i dati. Segui questi passaggi per eliminare il bucket Amazon S3.

1. Accedere alla console Amazon S3.
2. Nel riquadro di navigazione, seleziona Bucket.
3. Individua i <stack-name>bucket S3.
4. Seleziona il bucket S3 e scegli Elimina.

Per eliminare il bucket S3 utilizzando AWS CLI, esegui il comando seguente. Non è necessario prima svuotare il bucket quando si utilizza l'opzione --force.

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

## Eliminazione degli indici Amazon Kendra

Per prevenire la perdita accidentale di dati, questa soluzione è configurata per conservare gli indici Amazon Kendra creati dalla soluzione quando lo stack AWS è stato eliminato. CloudFormation Dopo aver disinstallato la soluzione, puoi eliminare manualmente gli indici Amazon Kendra per i quali non devi più conservare i dati. Segui questi passaggi per eliminare l'indice Amazon Kendra.

1. Accedi alla console Amazon Kendra.
2. Nel riquadro di navigazione, seleziona Indici.
3. Individua e seleziona l'indice che desideri eliminare.
4. Scegli Elimina per eliminare l’indice selezionato.

Per eliminare l'indice Amazon Kendra utilizzando l'interfaccia a riga di comando di AWS, esegui il seguente comando:

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

## Eliminazione dei log CloudWatch

Per prevenire la perdita accidentale dei dati, abbiamo configurato questa soluzione in modo da conservare CloudWatch i log se si decide di eliminare lo stack. CloudFormation Dopo aver disinstallato la soluzione, puoi eliminare manualmente i log se non hai bisogno di conservare i dati. Segui questi passaggi per eliminare i CloudWatch registri.

1. Accedi alla CloudWatch console Amazon.
2. Nel riquadro di navigazione, seleziona Log Groups.
3. Individua i gruppi di log creati dalla soluzione.
4. Seleziona uno dei gruppi di log.
5. Scegliere Actions (Operazioni), quindi selezionare Delete (Elimina VPC).

Ripetere i passaggi fino a eliminare tutti i gruppi di log della soluzione.

# Usa la soluzione

## Accesso all'interfaccia utente

Durante il processo di distribuzione dello stack (sia per la dashboard di distribuzione che per i casi d'uso) viene inviata un'e-mail all'indirizzo e-mail configurato. L'e-mail contiene le credenziali temporanee dell'utente che può utilizzare per registrarsi e accedere all'interfaccia web.

> **Note**
>
> L' DevOps utente con accesso alla Console di gestione AWS deve fornire all'utente amministratore l' CloudFront URL dell'interfaccia utente del dashboard di distribuzione al termine dello stack.

Per i casi d'uso, l'utente amministratore con accesso all'interfaccia utente del dashboard di distribuzione deve fornire all'utente aziendale l' CloudFront URL dell'interfaccia utente dello use case al termine della distribuzione.

Una volta effettuato l'accesso, l'utente può interagire con la soluzione UIs, utilizzando la dashboard di Deployment nel caso degli amministratori o lo use case nel caso degli utenti aziendali.

## Come aggiornare una distribuzione

Nella home page del dashboard di distribuzione (o nella pagina dei dettagli di una distribuzione) è possibile modificare la configurazione utilizzata da una distribuzione. È possibile modificare solo le distribuzioni che si trovano negli stati CREATE_COMPLETE o UPDATE_COMPLETE.

Ad eccezione del nome del caso d'uso, tutte le altre opzioni sono modificabili per una distribuzione. Basta modificare i valori che si desidera modificare e ridistribuire.

A seconda dell'ambito delle modifiche apportate, il tempo di ridistribuzione varierà. Potrebbero essere necessari alcuni secondi se sono state modificate impostazioni semplici (ad esempio, i parametri del modello), a più di 30 minuti se sono state modificate le opzioni relative all'infrastruttura più grandi (ad esempio, richiesta di creazione dell'indice Amazon Kendra per il Text use case RAG).

Una volta completata con successo la modifica, lo stato dell'applicazione riporterà lo stato UPDATE_COMPLETE. Al momento, puoi accedere all'interfaccia utente distribuita tramite l' CloudFront URL e interagire con la distribuzione modificata.

> **Note**
>
> Potrebbe essere più semplice eseguire più distribuzioni side-by-side se desideri confrontare impostazioni diverse o. LLMs Utilizza la funzionalità Clone per utilizzare rapidamente una configurazione esistente per avviare una nuova distribuzione.

## Come clonare una distribuzione

Nella home page del dashboard Deployments (o nella pagina dei dettagli di una distribuzione) è possibile clonare la configurazione utilizzata da una distribuzione. La clonazione di una distribuzione avvia la procedura guidata Deploy new use case, ma con la maggior parte dei campi precompilati con gli stessi valori.

Si tratta di un'operazione comoda che consente di duplicare rapidamente le distribuzioni con impostazioni modificate, ripristinare una distribuzione eliminata o confrontarne più di una in distribuzioni altrimenti identiche. LLMs

## Come eliminare una distribuzione

Nella home page del dashboard di Deployments (o nella pagina dei dettagli di una distribuzione), puoi eliminarla quando non è più necessaria la distribuzione. L'eliminazione di una distribuzione richiama un'operazione di eliminazione CloudFormation dello stack e predispone le risorse per la distribuzione.

Per impostazione predefinita, una distribuzione eliminata rimane ancora nella dashboard per abilitare la funzionalità di clonazione. Per rimuovere completamente una distribuzione dalla dashboard in modo che non venga più tracciata nell'interfaccia utente, scegli Elimina definitivamente nella finestra di conferma dell'eliminazione.

> **Important**
>
> Alcune risorse vengono lasciate indietro durante l'eliminazione dello stack e devono essere eliminate manualmente. Consulta la sezione Disinstallazione manuale per informazioni dettagliate su quali risorse vengono conservate e su come ripulirle.

# Configurazione di un Large Language Model (LLM)

Il LLM più adatto al tuo caso d'uso dipende da un'ampia serie di fattori specifici delle tue esigenze e dal tipo di esperienza del cliente che desideri curare. Questa soluzione non sembra prescrittiva, ma mira piuttosto a fornirti gli strumenti necessari per valutare ciò che funziona meglio per la tua applicazione.

Lo spazio generato dall'intelligenza artificiale si sta evolvendo rapidamente, quindi spetta a te tenerti aggiornato sui modelli, sulle tecniche di ottimizzazione e sulle migliori pratiche più recenti per assicurarti di creare le esperienze giuste per i tuoi clienti.

> **Note**
>
> Se lavori con dati non pubblici o sensibili, assicurati di selezionare un'opzione LLM utilizzando i servizi AWS (come Amazon Bedrock o Amazon SageMaker AI). Ciò migliora il livello di sicurezza generale della tua implementazione mantenendo i dati all'interno della tua regione e sulla rete AWS rispetto all'utilizzo di un LLM ospitato da un provider di terze parti.

## Utilizzo di Amazon SageMaker AI come provider LLM

A partire dalla versione 1.3.0, Amazon SageMaker AI è disponibile come fornitore di modelli per casi d'uso di testo. Questa funzionalità consente di utilizzare un endpoint di inferenza SageMaker AI già esistente all'interno dell'account AWS nella soluzione. Ecco alcuni modi per iniziare.

> **Important**
>
> La soluzione non gestisce il ciclo di vita degli endpoint SageMaker AI. Sei responsabile dell'eliminazione degli endpoint SageMaker AI una volta che non sono più necessari per evitare di incorrere in costi aggiuntivi.

### Creazione di un endpoint AI SageMaker

Puoi utilizzare Amazon SageMaker AI JumpStart per implementare rapidamente un endpoint.

Puoi anche utilizzare un endpoint SageMaker AI basato sulla generazione di testo e distribuirlo utilizzando il servizio AI di base. SageMaker Fai riferimento alla JumpStart documentazione

sull'SageMaker intelligenza artificiale per una guida dettagliata su come implementare un modello per l'inferenza.

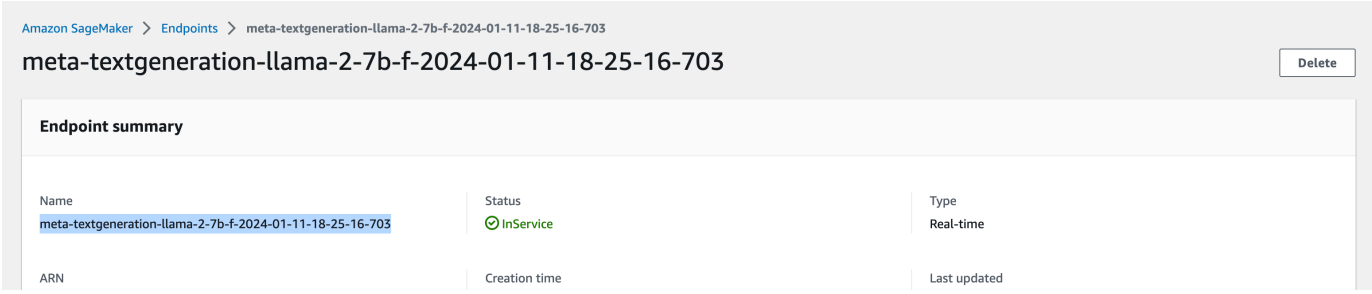
> **Note**
>
> models/LLMs Le basi sono in genere piuttosto grandi e spesso possono richiedere l'uso di istanze di calcolo accelerate di grandi dimensioni. Molte di queste istanze più grandi potrebbero non essere disponibili per impostazione predefinita nel tuo account AWS. Fai riferimento alle quote SageMaker AI predefinite e assicurati di richiedere un aumento della quota prima della distribuzione per evitare errori di distribuzione comuni.

## Usa un endpoint SageMaker AI per creare un'implementazione di casi d'uso in formato testo

Per implementare un nuovo use case di testo utilizzando un endpoint SageMaker AI per l'inferenza:

1. Crea un nuovo caso d'uso tramite la procedura guidata del dashboard di distribuzione e completa i moduli fino a raggiungere la pagina di selezione dei modelli.
2. Nella pagina Modelli, seleziona SageMaker AI come fornitore del modello. Questo genererà un modulo personalizzato che richiederà tre input chiave da parte dell'utente:
   - Il nome dell'endpoint SageMaker AI che desideri utilizzare. DevOps gli utenti possono ottenerlo dalla console AWS. Tieni presente che l'endpoint deve trovarsi nello stesso account e nella stessa regione in cui è distribuita la soluzione.

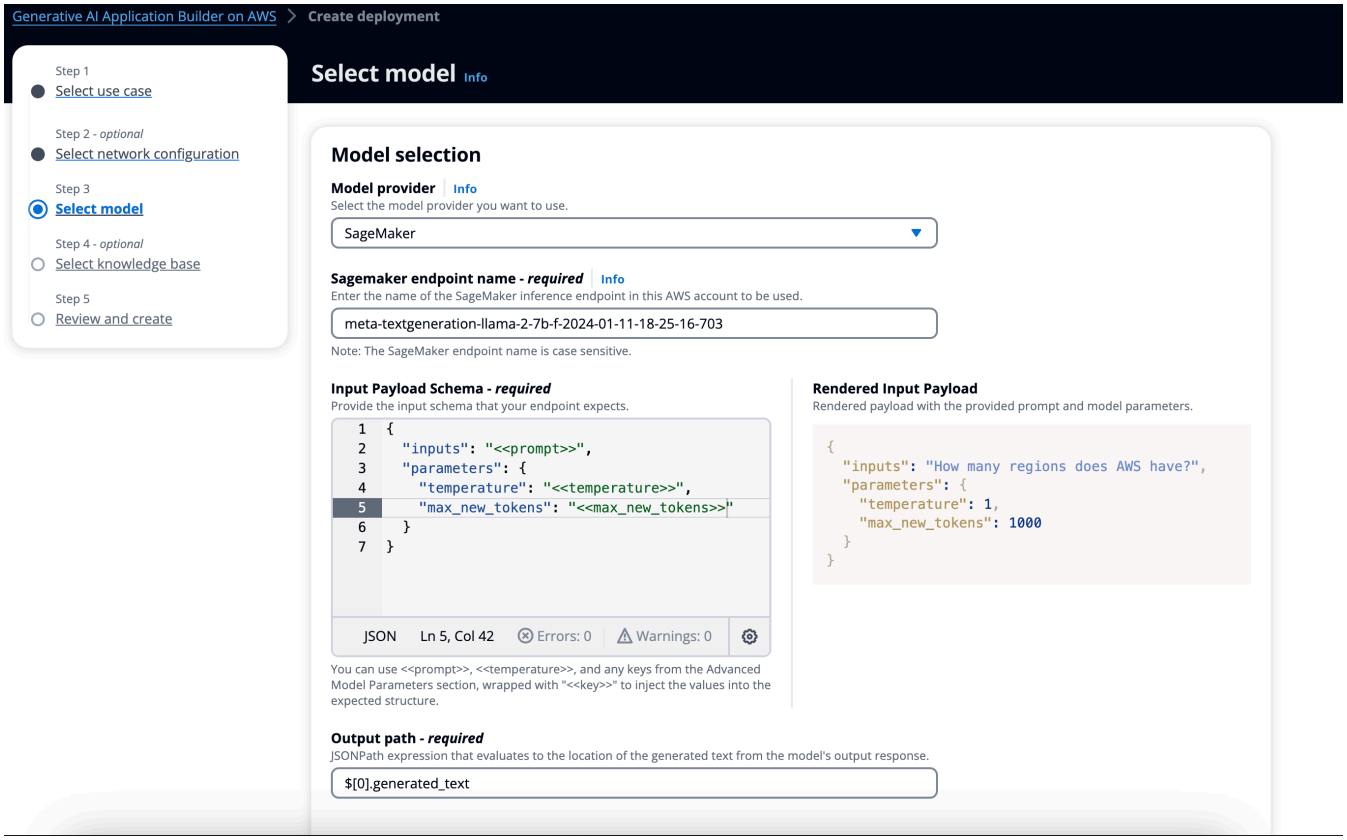
     Posizione del nome dell'endpoint sulla console AWS

   - Lo schema del payload di input previsto dall'endpoint. Per supportare il più ampio set di endpoint, gli utenti amministratori devono indicare alla soluzione in che modo l'endpoint prevede che l'input venga formattato. Nella procedura guidata di selezione del modello, fornite lo schema JSON per la soluzione da inviare all'endpoint. Puoi aggiungere segnaposto per inserire valori statici e dinamici nel payload della richiesta. Le opzioni disponibili sono:

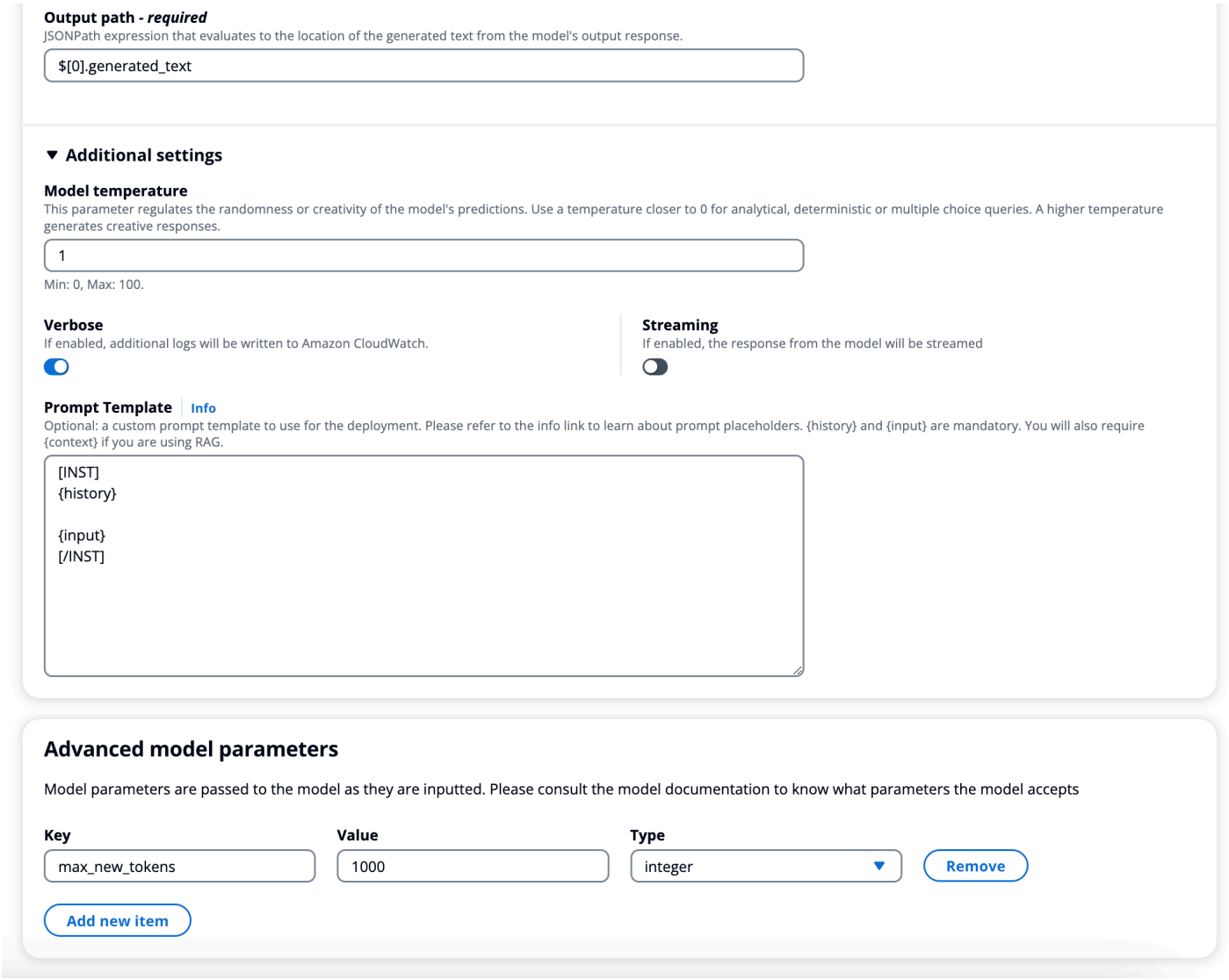
- Segnaposto obbligatori:\ <\ <prompt\ >\ > verranno sostituiti dinamicamente con l'input completo (ad esempio, cronologia, contesto e input dell'utente secondo il modello di prompt) da inviare all'endpoint AI in fase di esecuzione. SageMaker
- Segnaposto opzionali:\ <\ <temperature\ >\ > *,\ * e tutti i parametri definiti nei parametri del modello avanzato possono essere forniti all'endpoint. Qualsiasi stringa contenente un segnaposto racchiuso in\ <\ < and\ >\ > (ad esempio,\ <\ <max_new_tokens\ >\ >) verrà sostituita dal valore del parametro del modello avanzato con lo stesso nome.

  Schema di input di esempio: impostazione di campi obbligatori, prompt e temperatura, insieme a un parametro avanzato personalizzato, max_new_tokens. Il percorso di output deve essere fornito come stringa valida JSONPath




3. La posizione della risposta alla stringa LLMs generata all'interno del payload di output. Deve essere fornita come JSONPath espressione per indicare dove è previsto l'accesso alla risposta testuale finale mostrata agli utenti dall'interno dell'oggetto e della risposta di ritorno dell'endpoint.

   Esempio di aggiunta di parametri del modello Advanced da utilizzare all'interno dello schema di input SageMaker AI (vedere la Figura 2 per le opzioni/impostazioni precedenti)




> **Note**
>
> SageMaker L'intelligenza artificiale ora supporta l'hosting di più modelli sullo stesso endpoint e questa è la configurazione predefinita quando si implementa un endpoint nella versione corrente di SageMaker AI Studio (non Studio Classic).
> Se il tuo endpoint è configurato in questo modo, ti verrà richiesto di aggiungerlo `InferenceComponentName`alla sezione dei parametri del modello avanzato, con un valore corrispondente al nome del modello che desideri utilizzare.

# Impostazioni LLM avanzate

Durante l'utilizzo di Amazon Bedrock, puoi configurare alcune impostazioni avanzate per i tuoi modelli come Amazon Bedrock Guardrails, Provisioned Throughput for Amazon Bedrock e parametri di modello aggiuntivi.

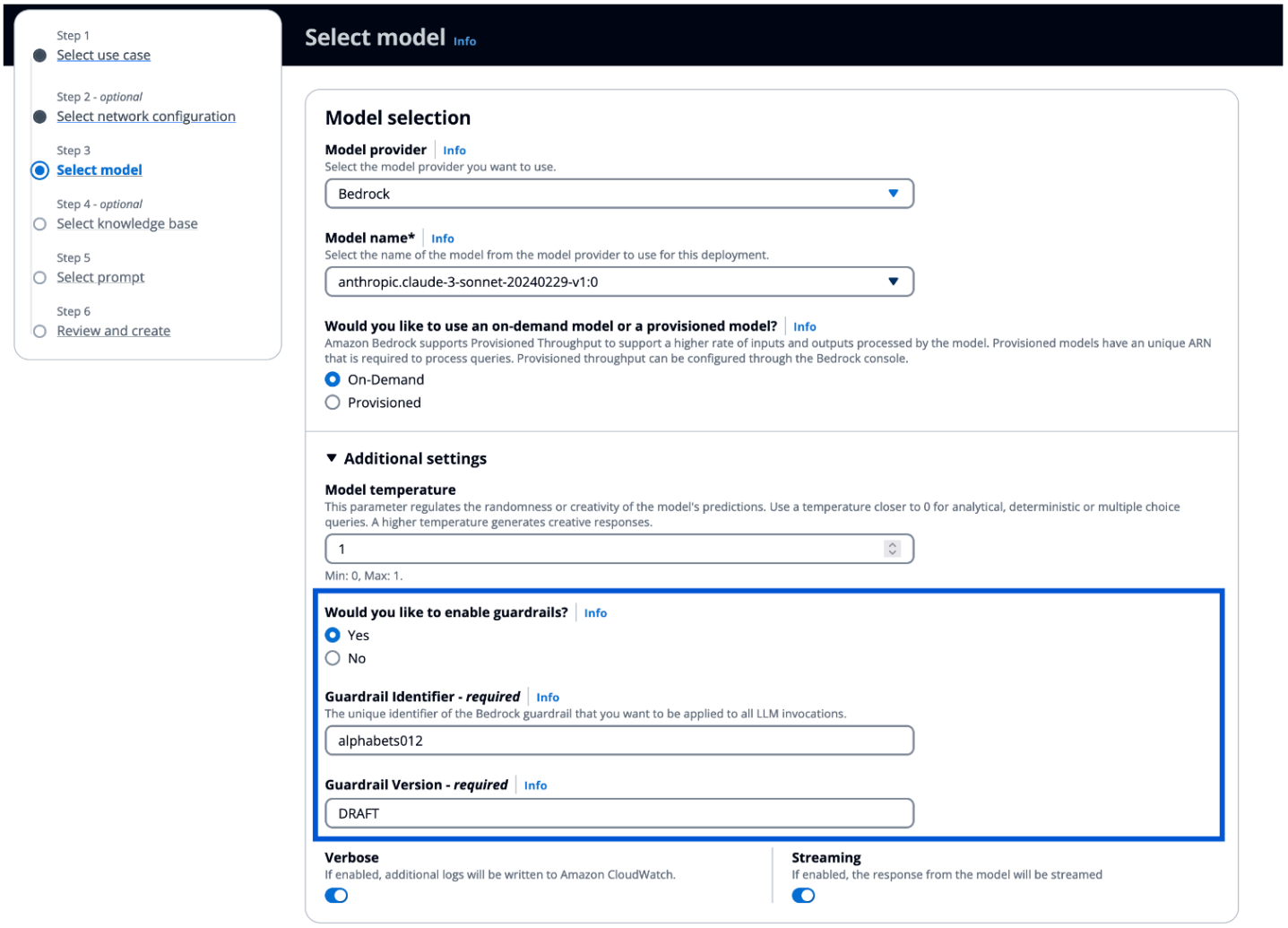
## Amazon Bedrock Guardrails

Amazon Bedrock Guardrails è una funzionalità di Amazon Bedrock che valuta gli input e le risposte LLM degli utenti in base a policy configurate dall'utente e fornisce un ulteriore livello di protezione, indipendentemente dal LLM sottostante selezionato dall'utente per un caso d'uso. Un Guardrail è composto da 2 politiche per evitare che i contenuti rientrino in categorie indesiderate o dannose:

1. Argomenti negati per definire una serie di argomenti indesiderati nel contesto dell'applicazione dell'utente, ad esempio, la consulenza in materia di investimenti in un'applicazione finanziaria e,
2. Filtri sui contenuti**** che consentono di filtrare i prompt di input degli utenti o le risposte dei modelli contenenti contenuti dannosi.

Per l'utilizzo nella soluzione Generative AI Application Builder, è necessario configurare un Guardrail nella console Amazon Bedrock utilizzando la procedura guidata Create guardrail. Una volta creato, puoi aggiungere questo Guardrail al caso d'uso della chat creato tramite la procedura guidata della soluzione Generative AI Application Builder nelle Impostazioni aggiuntive nella fase di selezione del modello fornendo la tua versione Guardrail Identifier e Guardrail.

Descrive la procedura guidata di distribuzione, che abilita Amazon Bedrock Guardrails



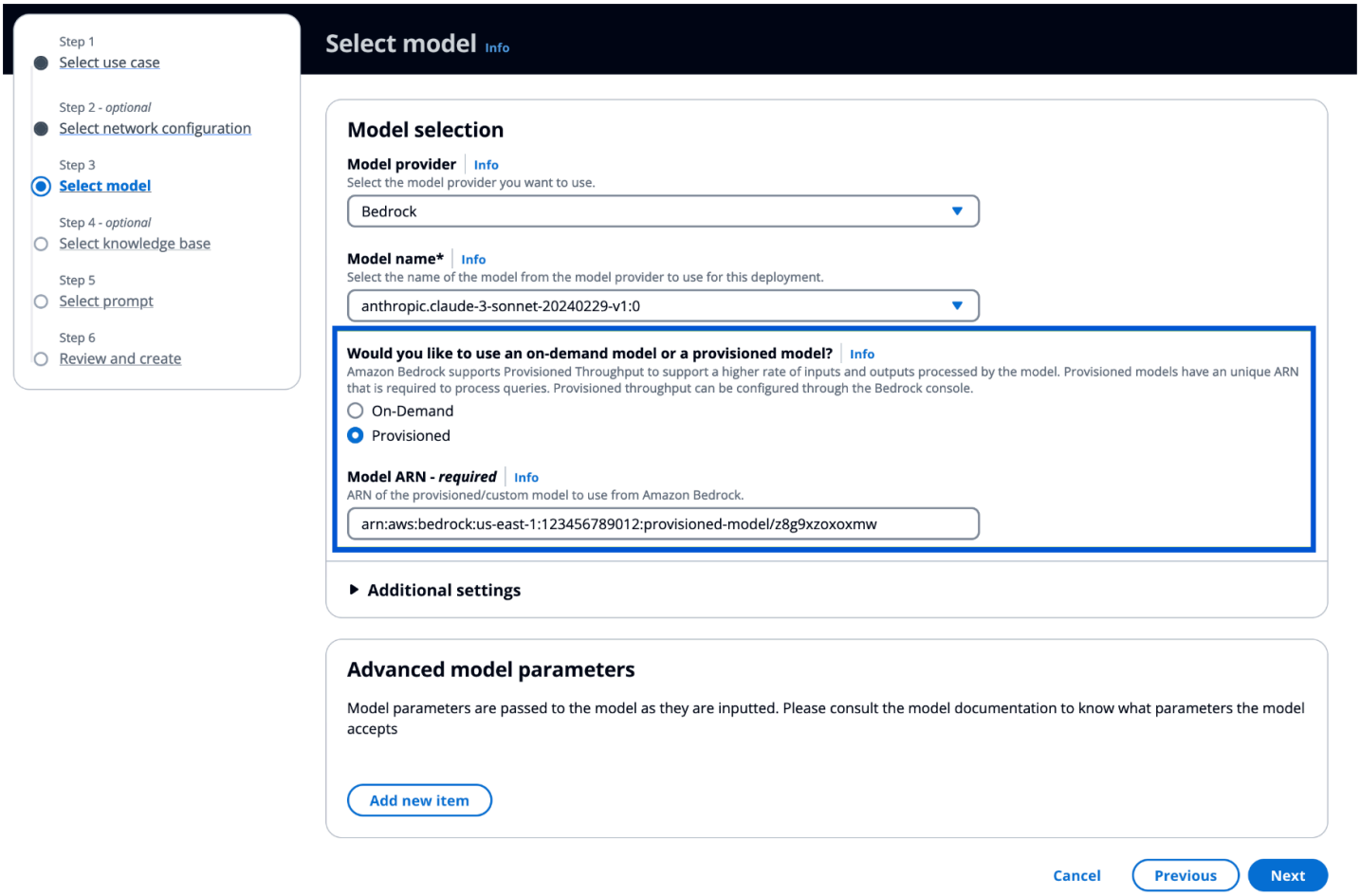

## Throughput assegnato per Amazon Bedrock

Ogni modello Amazon Bedrock on-demand segue il limite di quota di account specifico della regione per l'inferenza del modello. Ad esempio, Anthropic Claude 2.x su Bedrock attualmente consente l'elaborazione di 500 richieste e 500.000 token al minuto nelle regioni us-east-1 e us-west-2. Potresti anche voler utilizzare la soluzione con i tuoi modelli pre-addestrati perfezionati o continui. In questi casi, Amazon Bedrock consente un throughput assegnato che consente di eseguire carichi di lavoro di inferenza di grandi dimensioni e coerenti per modelli preaddestrati di base, ottimizzati o continui da utilizzare in applicazioni di produzione.

Una volta acquistato Provisioned Throughput all'interno della console Amazon Bedrock, viene generato un modello ARN per l'utilizzo. Ora puoi fornire questo Model ARN nella procedura guidata Generative AI Application Builder nella fase di selezione del modello. A tale scopo, seleziona Bedrock come fornitore del modello e il nome del modello di base utilizzato per generare questo Model ARN

fornito nella console Amazon Bedrock. Quindi, seleziona «Modello fornito» quando scegli tra modelli on demand e provisioned e fornisci il tuo Model ARN.

Descrive la procedura guidata di distribuzione: abilitazione del throughput fornito per Amazon Bedrock




> **Note**
>
> Il guardrail e il throughput assegnato devono trovarsi nella stessa regione del Deployment Dashboard e degli use case stack distribuiti.

## Parametri del modello

LLMs spesso accetta un'ampia gamma di parametri specifici per la sua implementazione. I fornitori di modelli forniscono spesso la documentazione che descrive l'insieme dei parametri supportati e i relativi utilizzi.

La soluzione passa i parametri del modello direttamente al modello sottostante, pertanto è importante assicurarsi che i parametri siano impostati correttamente. Consultate la documentazione del fornitore del modello per le informazioni più recenti sui parametri supportati.

# Configurazione di Agent Builder

Agent Builder offre opzioni di configurazione complete per la creazione di agenti AI pronti per la produzione. Questa sezione descrive come configurare e gestire le distribuzioni di Agent Builder.

## Configurazione del prompt di sistema

Il prompt di sistema definisce il comportamento, la personalità e le capacità dell'agente. Per configurare il prompt di sistema:

1. Nella procedura guidata di Agent Builder, vai al passaggio Configura agente.
2. Modifica il modello di prompt di sistema nell'editor di testo.
3. Includi istruzioni chiare per:
   - Ruolo e scopo dell'agente
   - Come utilizzare gli strumenti disponibili (server MCP)
   - Preferenze di formattazione delle risposte
   - Linee guida comportamentali
4. Se necessario, utilizzate il pulsante Ripristina le impostazioni predefinite per ripristinare il modello originale.

Procedure consigliate per i prompt degli agenti:

- Sii specifico sulle capacità e i limiti dell'agente
- Fornisci esempi chiari del comportamento desiderato
- Includi istruzioni per l'utilizzo degli strumenti e quando richiamarli
- Definisci le aspettative relative al formato di risposta
- Stabilisci dei limiti per il comportamento degli agenti

## Integrazione del server MCP

I server Model Context Protocol (MCP) forniscono agli agenti l'accesso a strumenti e fonti di dati aziendali. Per configurare i server MCP:

1. Nel passaggio Configure Agent, individua la sezione Server MCP.
2. Seleziona uno dei server MCP disponibili nel menu a discesa.

> **Note**
>
> I server MCP devono essere configurati e accessibili prima della distribuzione degli agenti. L'agente scoprirà e utilizzerà automaticamente gli strumenti esposti dai server MCP configurati. Fate riferimento alla documentazione MCP per la configurazione del server e degli strumenti.

## Impostazioni della memoria

Agent Builder fornisce due tipi di memoria per mantenere il contesto e le conoscenze:

Memoria a breve termine

Attivato per impostazione predefinita per tutti gli agenti:

- Mantiene il contesto della conversazione all'interno delle sessioni
- Acquisisce automaticamente i messaggi degli utenti e le risposte degli agenti
- Organizzato da actorID e sessionID per un corretto isolamento
- Nessuna configurazione richiesta

Memoria a lungo termine

Funzionalità opzionale per l'archiviazione di informazioni dettagliate tra le sessioni:

1. Nel passaggio Configure Agent, individua la sezione Configurazione della memoria.
2. Attiva Abilita l'attivazione della memoria a lungo termine.
3. Se abilitato, l'agente può:

- Estrarre e archiviare informazioni importanti tra le conversazioni
- Recupera il contesto pertinente dalle sessioni precedenti
- Acquisisci conoscenze sulle preferenze e sulla cronologia degli utenti

> **Note**
>
> La memoria a lungo termine utilizza AgentCore la memoria con una strategia di memoria semantica e impostazioni di conservazione predefinite.

## Monitoraggio delle implementazioni di Agent Builder

Agent Builder offre un monitoraggio completo tramite CloudWatch dashboard e metriche.

CloudWatch Accesso ai pannelli di controllo

1. Accedi alla CloudWatch console nel tuo account AWS.
2. Seleziona Dashboard dalla barra di navigazione a sinistra.
3. Trova il nome `AgentBuilder-<UseCaseId>` del pannello di controllo.
4. Visualizza le metriche in tempo reale e i dati storici sulle prestazioni.

Accesso e analisi dei log

I registri degli agenti sono disponibili nella sezione CloudWatch Registri:

1. Accedi a CloudWatch Logs nella console AWS.
2. Trova i gruppi di log con il prefisso. `/aws/bedrock-agentcore/runtimes/`
3. Usa CloudWatch Insights per interrogare e analizzare i log.
4. Cerca modelli di richiesta IDs o errore specifici.

# Configurazione di Workflow Builder

Workflow Builder consente l'orchestrazione di più agenti tramite un agente supervisore che delega il lavoro ad agenti Agent Builder specializzati.

## Creazione di un flusso di lavoro

1. Passa alla dashboard di distribuzione
2. Seleziona Crea un caso d'uso del flusso di lavoro
3. Configura l'agente supervisore:
   - Nome: nome descrittivo del flusso di lavoro
   - Descrizione: Scopo e funzionalità
   - System Prompt: istruzioni per la delega e il coordinamento degli agenti
   - Modello: Modello base per l'agente supervisore

Migliori pratiche per le richieste del supervisore:

- Descrivi chiaramente quando utilizzare ciascun agente specializzato
- Includi istruzioni per aggregare i risultati di più agenti
- Definisci le aspettative di formattazione delle risposte
- Stabilisci dei limiti per il comportamento di delega

## Selezione dell'agente

Seleziona gli agenti di Agent Builder da includere come agenti specializzati:

1. Fate clic su Aggiungi agente nella configurazione del flusso di lavoro
2. Sfoglia o cerca gli agenti Agent Builder disponibili
3. Consulta le descrizioni degli agenti
4. Seleziona gli agenti da includere nel flusso di lavoro

Descrizioni degli agenti

L'agente supervisore utilizza le descrizioni degli agenti per decidere a quale agente delegare. Assicurati che le descrizioni spieghino chiaramente:

- Dominio o funzionalità specializzati dell'agente
- Tipi di attività gestite dall'agente
- Aspettative di input/output

## Flussi di lavoro di test

Dopo la distribuzione:

1. Accedi al flusso di lavoro tramite il Deployment Dashboard
2. Esegui test con query che richiedono più agenti
3. Monitora la delega degli agenti nei registri CloudWatch
4. Esamina la qualità delle risposte e i modelli di delega
5. Modifica la richiesta del supervisore se la delega non è ottimale

# Suggerimenti per la gestione dei limiti dei token del modello

Nota: la soluzione non tenta direttamente di gestire i limiti dei token imposti da vari LLMs. Verifica e assicurati che la richiesta rimanga entro i limiti disponibili applicati dal fornitore del modello.

Per controllare la dimensione dei prompt, provate quanto segue:

1. Acquisisci familiarità con i limiti imposti dal modello che desideri utilizzare. Questi valori possono differire notevolmente tra i modelli, quindi è importante sapere qual è il budget disponibile prima di iniziare.
2. Crea il tuo prompt iniziale tenendo presente quel budget e considera quanto vuoi risparmiare per eventuali elementi dinamici del prompt. Ad esempio, input dell'utente, cronologia chat, estratti di documenti e così via.
3. Nella pagina di configurazione del prompt, imposta un limite per la dimensione della cronologia finale per limitare il numero di turni di conversazione inclusi nel prompt.
4. Imposta i limiti di restituzione dei documenti nella procedura guidata di configurazione della Knowledge Base. È necessario cercare di trovare il giusto equilibrio tra fornire all'LLM un contesto sufficiente per eseguire l'operazione, ma non tanto da superare i limiti dei token o influire negativamente sulla latenza.
5. Lascia un po' di buffer. Non pensate al budget per i casi tipici, ma pensate e sperimentate casi limite, come lunghe domande di input, estratti di documenti di grandi dimensioni o lunghe conversazioni.

# Passaggi per creare un server MCP Docker Image

Per utilizzare i server MCP (Model Context Protocol) con Generative AI Application Builder su AWS, come primo passaggio è necessaria un'immagine Docker creata e archiviata in un repository Amazon ECR privato.

> **Note**
>
> Al momento, i server MCP distribuiti esistenti nel AgentCore runtime di Amazon Bedrock non possono essere esportati in GAAB. Affinché i server MCP possano essere collegati agli agenti creati tramite GAAB, devono essere creati tramite GAAB.

## Fase 1: Crea il tuo server MCP

Innanzitutto, è necessario disporre dell'implementazione del server MCP pronta. Per istruzioni dettagliate sulla creazione di un server MCP, consulta la Amazon Bedrock AgentCore Developer Guide - Create an MCP server.

Consigliamo la seguente struttura di progetto:

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

Per la struttura Dockerfile, consigliamo di utilizzare un formato simile al seguente esempio:

```
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"]
```

## Fase 2: Esegui il test del server MCP a livello locale

Prima di eseguire la distribuzione su AWS, è importante testare il server MCP localmente per assicurarsi che funzioni come previsto. Per istruzioni dettagliate sui test locali, consulta la Amazon Bedrock AgentCore Developer Guide - Esegui il test del tuo server MCP localmente.

## Fase 3: Implementazione su Amazon ECR

Una volta creato e testato localmente il server MCP, segui questi passaggi per distribuirlo su Amazon ECR:

1. Assicurati di avere installato la versione più recente di AWS CLI e Docker. Per ulteriori informazioni, consulta Getting Started with Amazon ECR.
2. Recupera un token di autenticazione e autentica il client Docker nel registro. Usa l'interfaccia a riga di comando di AWS:

```
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. Crea la tua immagine Docker usando il seguente comando. Per informazioni sulla creazione di un file Docker da zero, consulta la documentazione Docker. Puoi saltare questo passaggio se l'immagine è già stata creata:

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

4. Una volta completata la compilazione, tagga l'immagine in modo da poterla inviare a questo repository:

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

5. Esegui il seguente comando per inviare questa immagine al tuo repository AWS appena creato:

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

Per istruzioni complete sulla distribuzione, consulta la Amazon Bedrock AgentCore Developer Guide - Deploy your MCP server to AWS.

### Fase 4: utilizza l'URI ECR in GAAB

Dopo aver inviato con successo l'immagine Docker ad Amazon ECR, copia l'URI dell'immagine dalla console ECR. Utilizzerai questo URI per distribuire il tuo server MCP tramite la procedura guidata di distribuzione Generative AI Application Builder on AWS.

## Passaggi per creare diversi obiettivi MCP Gateway

Amazon Bedrock AgentCore Gateway ti consente di trasformare i servizi AWS esistenti e APIs in strumenti MCP che possono essere utilizzati dai tuoi agenti. Il Gateway supporta diversi tipi di target, consentendoti di integrare diversi servizi di backend senza problemi.

Sono supportati i seguenti tipi di target:

- Obiettivi Lambda: trasforma le funzioni di AWS Lambda in strumenti MCP. Per istruzioni dettagliate, consulta la Amazon Bedrock AgentCore Developer Guide - Add Lambda targets.

- Obiettivi OpenAPI: utilizza le specifiche OpenAPI per definire ed esporre REST come strumenti MCP. APIs Per istruzioni dettagliate, consulta lo schema Amazon Bedrock AgentCore Developer Guide - OpenAPI.
- Obiettivi di Smithy: crea strumenti MCP utilizzando le definizioni dei modelli Smithy per integrazioni API sicure dai tipi. Per istruzioni dettagliate, consulta la Amazon Bedrock AgentCore Developer Guide - Building Smithy targets.
- Obiettivi del server MCP: Connettiti direttamente ai server MCP esterni tramite endpoint URL, consentendoti di integrare i server MCP esistenti. Per istruzioni dettagliate, consulta la Amazon Bedrock AgentCore Developer Guide - MCP servers targets.

Per ulteriori esempi e tutorial sulla creazione di obiettivi MCP Gateway, visita l'archivio di esempi di Amazon AgentCore Bedrock.

## Configurazione di una knowledge base

Questa sezione descrive come inserire dati nella knowledge base selezionata per la soluzione. La soluzione attualmente supporta Amazon Kendra e Amazon Bedrock Knowledge Base come knowledge base per la distribuzione di use case basati su RAG.

Amazon Kendra

Se utilizzi Amazon Kendra come knowledge base, consulta la Amazon Kendra Developer Guide per informazioni su come utilizzare vari connettori di sorgenti di dati per aiutarti a importare dati da un'ampia selezione di fonti.

Importante: per prevenire la perdita accidentale dei dati, la soluzione non elimina automaticamente l'indice Kendra (creato dalla soluzione o in altro modo) quando viene eliminata una distribuzione o uno stack. Se desideri eliminare la tua knowledge base ed evitare di incorrere in costi, consulta la sezione Disinstallazione manuale per i dettagli su quali risorse vengono conservate e su come ripulirle.

Basi di conoscenza di Amazon Bedrock

Le Knowledge Base di Amazon Bedrock possono essere supportate da una varietà di archivi vettoriali diversi, ciascuno con la capacità di indicizzare i dati. Per configurare e compilare la tua knowledge base, consulta la Amazon Bedrock User Guide. In particolare, ti consigliamo di:

- Per prima cosa configura la tua fonte di dati

- Quindi configura un indice vettoriale per la tua knowledge base in un archivio vettoriale supportato. Tieni presente che questo può essere ignorato se utilizzi l'opzione «Creazione rapida di un nuovo archivio vettoriale» nella console Bedrock durante la creazione della knowledge base.
- Infine, puoi creare la knowledge base e sincronizzare le fonti di dati configurate.

# Impostazioni avanzate della knowledge base

Le impostazioni avanzate della Knowledge Base come Knowledge Base Filtering e RAG con controllo degli accessi basato sui ruoli sono disponibili per l'uso con la soluzione. Il filtro della Knowledge Base può essere applicato a entrambe le Knowledge Base, mentre RAG con Role Based Access Control è disponibile specificamente per Amazon Kendra.

## Filtraggio della Knowledge Base

La soluzione consente di specificare i filtri degli attributi di Amazon Kendra o i filtri di recupero della knowledge base Bedrock durante la distribuzione di un caso d'uso nella sezione Configurazioni RAG avanzate del passaggio della knowledge base della procedura guidata. Questi filtri definiscono il modo in cui vengono richieste le fonti di dati nella knowledge base, ad esempio le strategie di ricerca, le lingue del documento sottostante oggetto delle query, ecc.

In entrambi i casi, viene utilizzato un oggetto JSON per specificare le impostazioni del filtro in base al formato specificato nella documentazione di ciascun servizio (come collegato sopra).

Esempio 1: Kendra AttributeFilter

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

Esempio 2: Bedrock RetrievalFilter

```
{
 "equals": {
 "key": "language",
```

```
  "value": "es"
  }
}
```

## RAG con controllo degli accessi basato sui ruoli con Amazon Kendra

Il controllo degli accessi basato sul ruolo (RBAC) consente di controllare quali utenti o gruppi possono accedere a determinati documenti nel tuo indice Amazon Kendra o visualizzare determinati documenti nei risultati di ricerca. Per configurare RBAC per il tuo Amazon Kendra Index ID con il tuo caso d'uso Generative AI Application Builder on AWS (GAAB), segui questi passaggi:

1. Configurazione dell'indice Amazon Kendra

1. Assicurati di aver creato un indice Amazon Kendra e di aver aggiunto almeno una fonte di dati.
2. Configura il controllo degli accessi per la tua fonte di dati in base ai gruppi di utenti. Per un'origine dati S3, segui le istruzioni disponibili nella documentazione per configurare gli elenchi di controllo degli accessi (ACLs) utilizzando gli stessi nomi di gruppo creati nel tuo pool di utenti Amazon Cognito. Ciò garantisce che gli utenti possano accedere solo ai documenti e ai risultati di ricerca che sono autorizzati a visualizzare in base all'appartenenza al gruppo.

> **Note**
>
> In Controllo dell'accesso utente nell'indice Kendra che hai creato, lascia il controllo dell'accesso utente basato su token su No. Quando abiliti il controllo degli accessi basato sui ruoli nella fase 2, Generative AI Application Builder su AWS estrae le dichiarazioni appropriate dal token di autenticazione dell'utente e crea un filtro degli attributi.

2. Implementate RAG Use Case utilizzando GAAB Deployment Wizard

1. Segui le istruzioni della procedura guidata sullo schermo in GAAB Deployment Wizard fino a raggiungere il passaggio 4 della procedura guidata per configurare RAG.
2. Nella fase Select Knowledge Base della procedura guidata di distribuzione, scegli Amazon Kendra come tipo di knowledge base.
3. Specificate se disponete di un indice Amazon Kendra esistente o se desiderate crearne uno nuovo. Se disponi di un indice esistente, fornisci l'ID del tuo indice Amazon Kendra che è stato configurato con liste di controllo degli accessi ACLs () basate su gruppi di utenti.

4. Abilita l'opzione Role Based Access Control. Questa opzione garantisce che i risultati di ricerca restituiti dall'indice Amazon Kendra vengano filtrati in base al ruolo dell'utente e alle autorizzazioni del gruppo.
5. Rivedi e implementa il caso d'uso.

3. Configurare Amazon Cognito

1. Individua il pool di utenti di Amazon Cognito utilizzato dalla tua distribuzione GAAB. Questo pool di utenti di Amazon Cognito viene in genere creato dallo stack di dashboard CloudFormation di distribuzione principale.
2. Crea nuovi utenti nel pool di utenti di Amazon Cognito. Durante la creazione di utenti, seleziona l'opzione «Invia un invito via e-mail» in modo che gli utenti ricevano le credenziali di accesso temporanee via e-mail. Ciò consente ai nuovi utenti di registrarsi e accedere all'applicazione GAAB.
3. Crea gruppi di utenti nel pool di utenti di Amazon Cognito. Assicurati che i nomi dei gruppi corrispondano esattamente ai gruppi configurati nel tuo indice Amazon ACLs Kendra. Questo è fondamentale per abilitare l'RBAC, poiché l'appartenenza al gruppo dell'utente determinerà i risultati della ricerca a cui potrà accedere.
4. Assegna gli utenti ai gruppi appropriati in base ai loro ruoli e alle autorizzazioni di accesso. Gli utenti devono essere aggiunti sia al gruppo richiesto per l'indice Amazon Kendra ACL, sia al gruppo specifico del caso d'uso creato durante la distribuzione GAAB. Ciò garantisce che gli utenti dispongano delle autorizzazioni necessarie per accedere al caso d'uso specifico e ai risultati di ricerca pertinenti.

Seguendo questi passaggi, avrete configurato il controllo degli accessi basato sui ruoli (RBAC) per la vostra implementazione GAAB, assicurando che gli utenti possano accedere e interagire solo con le informazioni e le funzionalità per cui sono autorizzati, in base al gruppo di utenti e alle autorizzazioni loro assegnati.

> **Note**
>
> Al momento, solo Amazon Kendra supporta RBAC per le knowledge base in Generative AI Application Builder su AWS. Per Amazon Bedrock Knowledge Base, RBAC non è supportato, ma puoi utilizzare filtri di metadati per ottenere un certo livello di filtraggio. Per ulteriori informazioni, consulta la Amazon Bedrock User Guide.

# Configurazione delle istruzioni

La procedura guidata del dashboard di distribuzione prevede una procedura di configurazione rapida che consente di personalizzare l'esperienza di richiesta e il modello che guiderà le interazioni tra gli utenti e il modello di intelligenza artificiale. La corretta configurazione di queste impostazioni è fondamentale per ottenere risposte accurate e pertinenti dall'assistente AI.

Questa sezione controlla l'esperienza e il comportamento complessivi del prompt AI.

- Lunghezza massima del modello di prompt: questa impostazione determina la lunghezza massima (in caratteri) del modello di prompt. Un valore più alto consente di fornire un contesto più ampio al modello di intelligenza artificiale, portando potenzialmente a risposte più accurate. Tuttavia, istruzioni eccessivamente lunghe possono anche causare rumore e influire negativamente sulle prestazioni. Per i modelli Amazon Bedrock, i valori predefiniti per la lunghezza massima del modello di prompt (in caratteri) vengono calcolati utilizzando i limiti del token del modello sottostante. Se modifichi e modifichi il nome di un modello all'interno di Bedrock, il pulsante «Ripristina impostazioni predefinite» viene evidenziato e può essere utilizzato per adottare le impostazioni predefinite del modello appena selezionato. Per i modelli Amazon SageMaker AI, vengono forniti valori predefiniti ragionevoli, ma si consiglia di controllare il modello sottostante e scegliere di conseguenza la lunghezza massima del modello di prompt e le lunghezze del testo di input. Per ulteriori informazioni, consulta la sezione Suggerimenti sulla gestione dei limiti dei token del modello.
- Lunghezza massima del testo di input: questa impostazione limita la lunghezza massima (in caratteri) del testo di input dell'utente. Gli input più lunghi possono contenere informazioni irrilevanti, aumentando il rischio di ottenere risposte irrilevanti o imprecise dal modello di intelligenza artificiale.
- Modifica dei prompt degli utenti: questa opzione consente di abilitare o disabilitare la possibilità per gli utenti di modificare il modello di prompt tramite l'interfaccia utente della chat. La disabilitazione di questa funzionalità può aiutare a mantenere la coerenza e prevenire modifiche involontarie al prompt.

### Modello di prompt

Questa sezione consente di definire il modello di prompt effettivo che verrà utilizzato dal modello AI. Il modello di prompt segue in genere una struttura che include segnaposto per vari componenti, come l'input dell'utente, i passaggi di riferimento e la cronologia chat.

- Modello di prompt: questa è l'area di testo principale in cui è possibile scrivere o incollare il modello di prompt desiderato. Il modello deve essere creato per fornire il contesto e le istruzioni necessari al modello di intelligenza artificiale. In genere include i seguenti segnaposto:
  - `{input}`: Questo segnaposto è obbligatorio per le implementazioni di intelligenza artificiale di Sagemaker e verrà sostituito con l'input o la query dell'utente.
  - `{history}`: Questo segnaposto è obbligatorio per le implementazioni di Sagemaker AI e verrà sostituito dalla cronologia chat della conversazione corrente.
  - `{context}`: Questo segnaposto è obbligatorio per le implementazioni RAG e verrà sostituito con gli estratti del documento ottenuti dalla knowledge base configurata.
- Riformulare la domanda? : Questa opzione (disponibile solo per le implementazioni RAG) determina se la query di input originale dell'utente deve essere riformulata o disambiguata prima di essere passata al modello di intelligenza artificiale. La riformulazione della query a volte può aiutare il modello a comprendere meglio l'intento dell'utente, portando potenzialmente a risposte più accurate.

Quando si configurano il modello di prompt e l'esperienza, è essenziale trovare un equilibrio tra fornire contesto e istruzioni sufficienti al modello di intelligenza artificiale ed evitare informazioni eccessivamente lunghe o irrilevanti che potrebbero introdurre problemi di rumore o prestazioni.

Impostazioni avanzate dei prompt

Questa sezione consente di controllare il modo in cui la cronologia delle conversazioni viene presentata al modello di intelligenza artificiale.

- Dimensione della cronologia finale: questa impostazione determina il numero di messaggi precedenti da includere nel prompt finale. L'impostazione di questo valore su zero comporterebbe l'immissione della cronologia né nel modello di prompt né nel modello di prompt di disambiguazione. Nota: anche se impostato su zero, è comunque necessario che nei modelli di prompt sia presente un segnaposto {history}. In fase di esecuzione, verrà sostituita con una stringa vuota.
  - Nota: si consiglia di fornire un numero pari per questo valore. L'immissione di un numero dispari comporterebbe la restituzione solo della risposta AI di un'interazione accoppiata.
- Prefisso umano: questo è il prefisso utilizzato per identificare i messaggi inviati dall'utente nella cronologia delle conversazioni.
- Prefisso AI: questo è il prefisso utilizzato per identificare i messaggi restituiti dal modello AI nella cronologia delle conversazioni.

Configurazione del prompt di disambiguazione

Questa sezione consente di configurare il comportamento e il modello per disambiguare gli input degli utenti prima di inviarli alla knowledge base configurata.

- Abilita la disambiguazione: questa opzione determina se gli input dell'utente devono essere disambiguati prima di inviarli alla knowledge base.
- Modello di richiesta di disambiguazione: questo è il modello di richiesta utilizzato per chiarire gli input dell'utente quando si è connessi a una knowledge base. L'output generato da questo prompt verrà utilizzato come interrogazione inviata alla knowledge base. La disabilitazione della disambiguazione comporterebbe l'invio invariato della query non elaborata dell'utente alla knowledge base.

Ad esempio, con la disambiguazione abilitata, una domanda utente successiva del tipo «Quanto costa?» potrebbe essere disambiguato in «Quanto costa rinnovare la mia targa?» , portando a una query di ricerca migliore.

# Utilizzo del caso d'uso Text distribuito

L'interfaccia utente integrata per lo use case Text ha lo scopo di consentire agli utenti aziendali di esplorare e sperimentare rapidamente la distribuzione creata dall'utente amministratore. Le modifiche alla configurazione apportate dall'utente aziendale hanno effetto solo per la sessione. L'utente aziendale deve condividere queste modifiche con l'utente amministratore che può aggiornare la distribuzione di base con tali modifiche affinché tutti possano utilizzarle.

L'interfaccia utente della chat è composta dai seguenti componenti:

- Finestra di chat
- Casella di immissione della chat
- Settings
- Conversazione chiara

## Finestra di chat

Mantiene diversi turni della conversazione. I messaggi che iniziano a destra provengono dall'utente aziendale, mentre i messaggi che iniziano a sinistra provengono dal LLM configurato. Una piccola icona con gli appunti è presente su tutte le risposte LLM per consentire una facile copia delle risposte.

## Casella di input per la chat

Nella parte inferiore della finestra di chat si trova la casella di immissione della chat. Qui gli utenti aziendali possono inserire i propri messaggi da inviare al LLM. Appena sopra la casella di input c'è lo stato della connessione. Se la connessione viene interrotta (ad esempio, a causa di inattività), al successivo invio di un messaggio di chat viene creata automaticamente una nuova connessione. Si prevede che questa richiesta richieda un po' più di tempo a causa del tempo di WebSocket connessione aggiuntivo.

In base alla configurazione specifica, potrebbe essere applicata una lunghezza massima all'input. Se questo limite viene superato, gli utenti ricevono un avviso e il messaggio non viene inviato.

Nota: se utilizzi RAG con Amazon Kendra, l'API Retrieve troncherà le query a 30 parole chiave. Se ti aspetti input più lunghi da parte dell'utente, valuta in che modo ciò potrebbe influire sulle prestazioni di ricerca.

## Settings

Per consentire agli utenti aziendali di sperimentare rapidamente diverse configurazioni, è disponibile un pannello delle impostazioni che consente di on-the-fly modificare determinate opzioni di configurazione di distribuzione

(esempio, modello di prompt). Queste modifiche possono essere apportate solo all'inizio di una nuova sessione. Una volta avviata una conversazione, la cancellazione della conversazione riattiva la modifica delle impostazioni di configurazione.

Nota: gli utenti amministratori possono scegliere di bloccare le impostazioni di una distribuzione. Possono impedire modifiche in tempo reale al momento della distribuzione tramite la procedura guidata durante la fase richiesta.

## Conversazione chiara

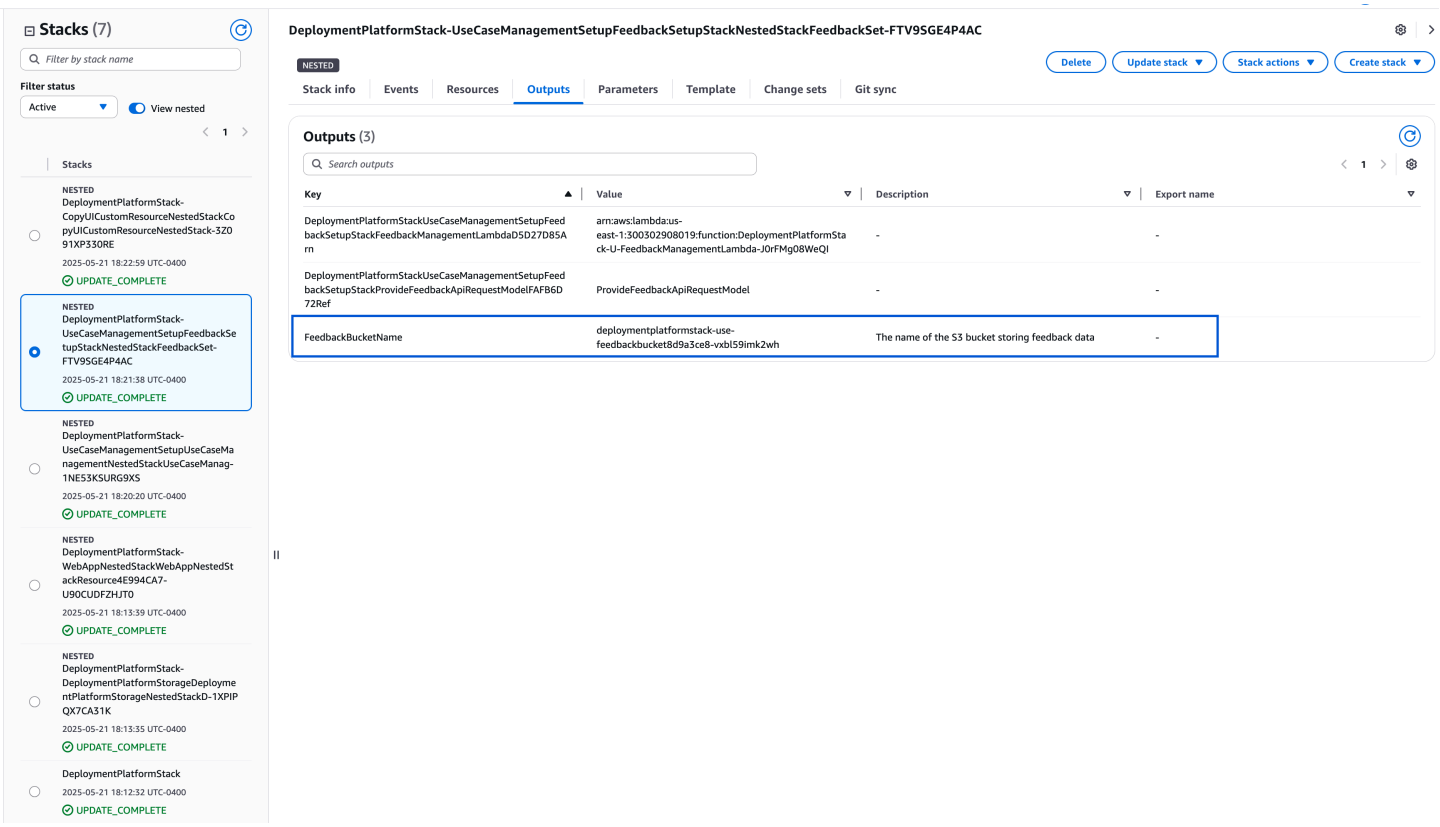
Nel corso della conversazione, la soluzione mantiene una cronologia delle chat, che consente un'esperienza di conversazione. Ciò consente la disambiguazione delle interrogazioni e le domande di follow-up. Per reimpostare una conversazione ed eliminare tutta la cronologia chat relativa a questa interazione, scegli *Cancella conversazione*nella parte superiore della finestra della chat. Una volta terminata la conversazione, viene creata una nuova sessione che riattiva la modifica delle impostazioni.

# Accesso e analisi del feedback raccolto dagli utenti

A partire dalla versione 3.0.0, Deployment Dashboard implementa uno stack di feedback annidato che consente ai casi d'uso di Text and Bedrock Agent distribuiti con il Dashboard di disporre della funzionalità di raccolta dei feedback per le risposte che generano. LLM/Agent In particolare, gli utenti possono fornire un feedback positivo o negativo insieme a un commento opzionale. Se l'utente fornisce un feedback negativo, può selezionare ulteriormente una di queste categorie negative: «Impreciso», «Incompleto o insufficiente», «Dannoso», «Altro». and/or

Una volta che l'utente ha fornito il feedback, il feedback viene archiviato in un bucket S3 partizionato per Use Case ID, anno e mese. L'Use Case ID si trova nella Deployment Dashboard e il bucket Feedback S3 si trova negli output dello stack annidato di feedback dello stack Deployment Dashboard:

Rappresenta lo stack di distribuzione - Finding Feedback Bucket Name




Il feedback degli utenti viene inviato come richiesta API contenente un set minimo di informazioni:

```
{
  "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."
}
```

Questo payload viene quindi elaborato da un lambda utilizzando il `useCaseRecordKey` che identifica la configurazione corretta di un caso d'uso al momento dell'implementazione. Questa configurazione viene utilizzata per ottenere dettagli specifici per il feedback, come il ConversationTable nome (contiene tutte le conversazioni e le sequenze di messaggi umani e di intelligenza artificiale), che viene ulteriormente utilizzato per recuperare l'effettivo e. `userInput` `llmResponse` A questo record di feedback vengono inoltre allegati dettagli aggiuntivi, ad esempio il caso d'uso `agentId` e `agentAliasId` per un caso d'uso Bedrock Agent e, ecc. `modelProviderbedrockModelId`, per un caso d'uso Text che utilizza questa configurazione. Per i dettagli su come accedere a questa configurazione, consulta la sezione Mappature di feedback personalizzate di seguito. Ogni richiesta di feedback in arrivo viene archiviata come oggetto JSON e un record di feedback di esempio può essere simile al seguente per un caso d'uso di tipo Text:

```
{
   "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"
}
```

o come questo per un caso d'uso di Bedrock Agent:

```
{
   "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"
}
```

Questo feedback può quindi essere utilizzato per ulteriori elaborazioni, analisi e modellazione di cicli di riformazione/feedback. È inoltre possibile aggiungere mappature personalizzate per migliorare il record di feedback archiviato nella lambda di feedback.

# Mappature di feedback personalizzate

Il Deployment Dashboard contiene un file `LLMConfigTable` che può essere trovato negli output dello stack Deployment Dashboard con la chiave. `LLMConfigTableName` `LLMConfigTable`contiene le configurazioni per ogni caso d'uso in base alle impostazioni selezionate dall'amministratore durante la distribuzione dello usecase tramite la procedura guidata di Deployment Dashboard. Ogni configurazione del caso d'uso è identificata dalla relativa. `useCaseRecordKey` Ecco un esempio di record di configurazione del caso d'uso in: `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"
   }
}
```

Se il feedback è abilitato per un caso d'uso, questa configurazione conterrà un `FeedbackParams` oggetto che consente di inserire al suo interno un `CustomMappings` oggetto in grado di specificare tutti JSONPaths i campi aggiuntivi da aggiungere al record JSON di feedback memorizzato nel bucket di feedback S3. Ad esempio, per la configurazione del caso d'uso di esempio precedente, CustomMappings contiene `NumberOfDocs` e `ScoreThreshold` JSONPaths inoltre contiene l'`CustomMappings`oggetto che inizia con la radice `config` di. JSONPath Con questa configurazione, ogni record JSON archiviato nel bucket di feedback S3 inizierà a ricevere questi 2 valori aggiuntivi oltre ai campi che sono già stati forniti.

## Analisi dei dati di feedback

I dati di feedback vengono archiviati in S3 come oggetti JSON. Ecco alcuni approcci per rendere questi dati di feedback più accessibili e utilizzabili:

### Utilizzo di AWS Glue e Amazon Athena

AWS Glue e Amazon Athena offrono un modo serverless per catalogare, interrogare e analizzare i dati di feedback.

AWS Glue ti consente di creare un crawler AWS Glue che ispeziona i dati in un bucket S3, ne deduce lo schema e registra tutti i metadati pertinenti in un catalogo. Successivamente, è possibile utilizzare servizi come Amazon Athena per interrogare i dati.

Puoi consultare la documentazione di AWS Athena sui passaggi per connettere il bucket S3 di feedback con Amazon Athena utilizzando AWS Glue Data Catalog. Puoi anche utilizzare alcune delle funzionalità più potenti di Glue per eseguire processi Extract Transform & Load (ETL) su questi dati e trasformarli in un formato adatto ai tuoi casi d'uso di analisi o riqualificazione dei modelli. Con Glue, puoi eseguire operazioni come filtrare i record con determinati tipi di feedback, compilare eventuali

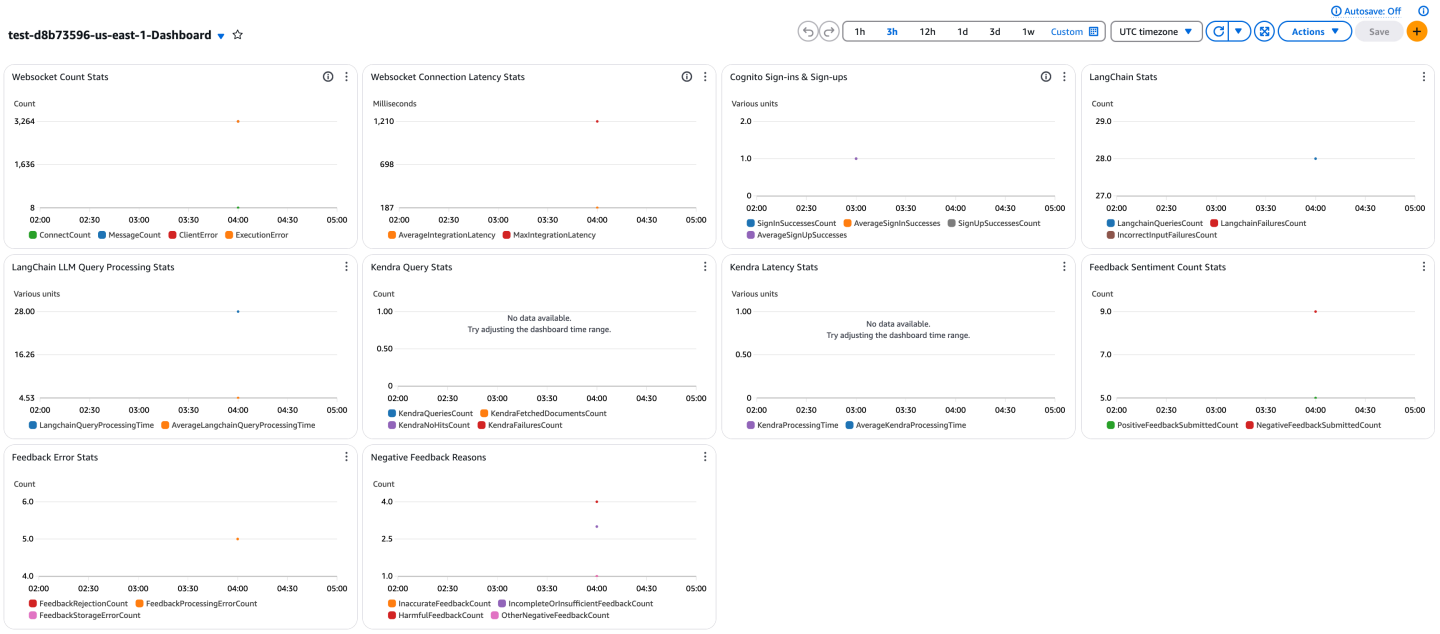
informazioni mancanti e puoi anche caricare questi dati in un'altra posizione di archiviazione come un altro bucket S3 o un altro data store AWS.

> **Note**
>
> A seconda del caso d'uso, valuta la possibilità di programmare l'esecuzione periodica del crawler Glue (ad esempio settimanalmente) anziché ogni notte per ottimizzare i costi, poiché i dati di feedback possono essere scarsi.

## CloudWatch Utilizzo delle dashboard della soluzione

È inoltre possibile accedere a un CloudWatch pannello di controllo fornito con la soluzione, in grado di fornire le tendenze relative ai feedback positivi e negativi, le categorie dei motivi dei feedback negativi e così via, in base ai singoli casi d'uso. Puoi trovare questa dashboard utilizzando il nome del tuo caso d'uso in Dashboards all'interno della console CloudWatch AWS:

Rappresenta Usecase Dashboard CloudWatch




Puoi anche creare widget aggiuntivi in questa dashboard o creare dashboard Amazon Quick Sight.

## Le migliori pratiche per l'analisi dei dati di feedback

- Implementa politiche relative al ciclo di vita dei dati sul tuo bucket S3 per archiviare i dati di feedback più vecchi su livelli di storage più economici

- Crea analisi separate per ogni caso d'uso per identificare opportunità di miglioramento specifiche del modello
- Stabilisci soglie di feedback che attivino avvisi quando il feedback negativo supera i livelli accettabili
- Esporta periodicamente informazioni importanti per condividerle con le parti interessate e i team di miglioramento dei modelli

## Visualizzazione delle metriche operative per una distribuzione

La dashboard di implementazione e gli stack di casi d'uso sono dotati ciascuno di un proprio CloudWatch dashboard che tiene traccia di varie metriche operative della soluzione. Puoi utilizzare queste CloudWatch dashboard per confrontare diverse implementazioni. Per accedere ai dashboard:

1. Passare alla console CloudWatch .
2. Cerca i dashboard predefiniti cercando il nome dello stack o l'identificatore univoco universale (UUID).

Ad esempio, lo use case Text include grafici che tracciano il numero di WebSocket connessioni, il numero di accessi e iscrizioni degli utenti, il tempo impiegato dall'LLM per elaborare un completamento e così via. I clienti possono utilizzare questi grafici per confrontare varie metriche _quantitative di una distribuzione.

Example

È difficile confrontare i risultati qualitativi di vari modelli applicati a diversi casi d'uso. Usa la funzione Clone per avviare rapidamente più implementazioni in modo da poter confrontare gli output fianco a fianco.

## Accedi a Logs Insights CloudWatch

Questa soluzione registra i messaggi di errore, di avviso, informativi e di debug per le funzioni Lambda. Per scegliere il tipo di messaggi da registrare:

1. Individua la funzione applicabile nella console AWS Lambda.
2. Aggiungi una variabile di ambiente POWERTOOLS_LOG_LEVEL.
3. Imposta la variabile sul tipo di messaggio applicabile.

Per ulteriori istruzioni, consulta Create Lambda Environmental Variables nella AWS Lambda Developer Guide.

La tabella seguente elenca i tipi di livelli di log tra cui puoi scegliere.

| Livello | Description |
| --- | --- |
| ERROR (ERRORE) | I log includono informazioni su tutto ciò che causa il fallimento di un'operazione. |
| ATTENZIONE | I log includono informazioni su tutto ciò che potrebbe causare incoerenze nella funzione ma non necessariamente causare il falliment o dell'operazione. I log includono anche messaggi ERROR. |
| INFORMAZIONI | I log includono informazioni di alto livello sul funzionamento della funzione. I log includono anche messaggi di ERRORE e AVVISO. |
| ESEGUIRE IL DEBUG | I log includono informazioni che potrebbero essere utili per il debug di un problema relativo alla funzione. I log includono anche i messaggi ERROR, WARNING e INFO. |

Utilizzare la procedura seguente per aggiungere CloudWatch Logs Insights a questa soluzione.

1. Identifica i gruppi di log pertinenti:
   a. Accedi alla CloudFormation console AWS.
   b. Scegli lo stack di destinazione.
   c. Seleziona la scheda Risorse e cerca le funzioni Lambda di destinazione.
   d. Accedi alla console AWS Lambda e scegli ciascuna delle funzioni Lambda di destinazione.
   e. Per ciascuna delle funzioni Lambda di destinazione, seleziona la scheda Monitor e scegli Visualizza CloudWatch registri.
   f. Copia i nomi dei gruppi di log da cui vuoi estrarre informazioni.
2. Accedi alla CloudWatch console Amazon.

3. Nel menu di navigazione, in Logs, scegli Logs Insights.
4. Nella pagina Logs Insights, scegli la scheda Logs.
5. Cerca i nomi dei gruppi di log dal passaggio 1.
6. Copia una delle seguenti query di esempio e incollala nel campo della query:
   a. Per identificare tutte le eccezioni relative al client:

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

   b. Per recuperare il numero di chiamate in base al nome della funzione:

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

   c. Per recuperare il conteggio delle chiamate a intervalli di cinque minuti:

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

   d. Per recuperare tutte le tracce di AWS IDs X-Ray:

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

   e. Per recuperare i log relativi a uno specifico X-Ray Trace ID:

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

   f. Per recuperare errori non autorizzati: WebSocket

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

   g. Per recuperare il conteggio delle metriche pubblicate:

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

# Guida per sviluppatori

Questa sezione fornisce il codice sorgente della soluzione, una guida all'integrazione, una guida allapersonalizzazione e un riferimento all'API.

## Codice sorgente

Visita il nostro GitHub repository per scaricare i file sorgente di questa soluzione e condividere le tue personalizzazioni con altri.

I modelli Generative AI Application Builder on AWS vengono generati utilizzando l'AWS Cloud Development Kit (AWS CDK). Consulta il file README.md per ulteriori informazioni.

## Guida all'integrazione

L'intera soluzione è progettata per essere facilmente estensibile. Il livello di orchestrazione di questa soluzione è costruito utilizzando. LangChain È possibile aggiungere qualsiasi fornitore di modelli, knowledge base o tipo di memoria di conversazione supportato da LangChain (o da una terza parte che fornisce LangChain connettori per questi componenti) a questa soluzione.

### Espansione supportata LLMs

Per aggiungere un altro fornitore di modelli, ad esempio un provider LLM personalizzato, è necessario aggiornare i seguenti tre componenti della soluzione:

1. Crea un nuovo stack `TextUseCase` CDK, che distribuisce l'applicazione di chat configurata con il tuo provider LLM personalizzato:
   a. Clona il GitHub repository di questa soluzione e configura il tuo ambiente di compilazione seguendo le istruzioni fornite nel file README.md.
   b. Copia (o creane uno nuovo), incollalo nella stessa directory e rinominalo in. `source/infrastructure/lib/bedrock-chat-stack.ts` `custom-chat-stack.ts`
   c. Rinomina la classe nel file con una classe adatta, ad esempio. `CustomLLMChat`
   d. Puoi scegliere di aggiungere un segreto di Secrets Manager a questo stack, che memorizza le tue credenziali per il tuo LLM personalizzato. È possibile recuperare queste credenziali durante l'invocazione del modello nel livello di chat Lambda discusso nel paragrafo successivo.

2. Crea e collega un layer Lambda contenente la libreria Python del provider di modelli da aggiungere. Per un'applicazione di chat per casi d'uso di Amazon Bedrock, la libreria `langchain-aws` Python contiene i connettori personalizzati nella parte superiore LangChain del pacchetto per connettersi ai provider di modelli AWS (Amazon Bedrock SageMaker e AI), alle knowledge base (Amazon Kendra e Amazon Bedrock Knowledge Bases) e ai tipi di memoria (come DynamoDB). Analogamente, altri fornitori di modelli dispongono di connettori propri. Questo livello consente di collegare la libreria Python di questo fornitore di modelli in modo da poter utilizzare questi connettori nel livello di chat Lambda, che richiama l'LLM (passaggio 3). In questa soluzione, viene utilizzato un bundler di asset personalizzato per creare livelli Lambda, collegati utilizzando aspetti CDK. Per creare un nuovo livello per la libreria Custom Model Provider:

   a. Passate alla `LambdaAspects` classe nel `source/infrastructure/lib/utils/lambda-aspects.ts` file.

   b. Segui le istruzioni su come estendere la funzionalità della classe di aspetti Lambda fornita nel file (ad esempio aggiungere il `getOrCreateLangchainLayer` metodo). Per utilizzare questo nuovo metodo (ad esempio,`getOrCreateCustomLLMLayer`), aggiorna anche l'`LLM_LIBRARY_LAYER_TYPE`Senum nel `source/infrastructure/lib/utils/constants.ts` file.

3. Estendi la funzione `chat` Lambda per implementare un builder, un client e un gestore per il nuovo provider.

   `source/lambda/chat`Contiene le LangChain connessioni per diverse classi LLMs insieme alle classi di supporto per crearle. LLMs Queste classi di supporto seguono i modelli di progettazione Builder e Object Oriented per creare l'LLM.

   Ogni gestore (ad esempio`bedrock_handler.py`) crea prima un client, controlla l'ambiente per le variabili di ambiente richieste e quindi chiama un `get_model` metodo per ottenere la LangChain classe LLM. Il metodo generate viene quindi chiamato per richiamare l'LLM e ottenere la sua risposta. LangChain attualmente supporta la funzionalità di streaming per Amazon Bedrock, ma non l' SageMaker intelligenza artificiale. In base alla funzionalità di streaming o non streaming, viene chiamato il WebSocket gestore (`WebsocketStreamingCallbackHandler`o`WebsocketHandler`) appropriato per inviare la risposta alla WebSocket connessione utilizzando il metodo. `post_to_connection`

   La `clients/builder` cartella contiene le classi che aiutano a creare un LLM Builder utilizzando il pattern Builder. Innanzitutto, a `use_case_config` viene recuperato da un archivio di configurazioni DynamoDB, che memorizza i dettagli sul tipo di knowledge base, memoria di conversazione e modello da costruire. Contiene anche dettagli rilevanti del modello, come i

parametri e i prompt del modello. Il Builder aiuta quindi a seguire i passaggi per la creazione di una knowledge base, la creazione di una memoria di conversazione per mantenere il contesto di conversazione per LLM, l'impostazione dei LangChain callback appropriati per i casi di streaming e non streaming e la creazione di un modello LLM basato sulle configurazioni del modello fornite. La configurazione di DynamoDB viene archiviata al momento della creazione del caso d'uso quando si distribuisce un caso d'uso dalla dashboard di distribuzione (o quando viene fornito dagli utenti nelle distribuzioni standalone dello stack di use case senza il dashboard di distribuzione).

La `clients/factories` sottocartella consente di impostare la memoria di conversazione e la classe di knowledge base appropriate, in base alla configurazione LLM. Ciò consente una facile estensione a qualsiasi altra knowledge base o tipo di memoria che si desidera supportare dall'implementazione.

La `shared` sottocartella contiene implementazioni specifiche della knowledge base e della memoria di conversazione, che vengono istanziate all'interno delle factory dal builder. Contiene anche i retriever Amazon Kendra e Amazon Bedrock Knowledge Base LangChain chiamati all'interno per recuperare i documenti per i casi d'uso RAG, insieme ai callback, utilizzati dal modello LLM. LangChain

Le LangChain implementazioni utilizzano LangChain Expression Language (LCEL) per comporre insieme catene di conversazioni. `RunnableWithMessageHistory`la classe viene utilizzata per mantenere la cronologia delle conversazioni con catene LCEL personalizzate, abilitando funzionalità come la restituzione di documenti sorgente e l'utilizzo della domanda riformulata (o disambigua) inviata alla knowledge base per essere inviata anche al LLM.

Per creare la propria implementazione di un provider personalizzato, è possibile:

a. Copia il `bedrock_handler.py` file e crea il tuo gestore personalizzato (ad esempio,`custom_handler.py`), che crea il tuo client personalizzato (ad esempio,`CustomProviderClient`) (specificato nel passaggio seguente).
b. Copia `bedrock_client.py` nella cartella client. Rinominalo in `custom_provider_client.py` (o il nome del fornitore del modello specifico, ad esempio`CustomProvider`). Assegna un nome appropriato alla classe al suo interno, ad esempio la classe `CustomProviderClient` che eredita. `LLMChatClient`

   È possibile utilizzare i metodi forniti da `LLMChatClient` o scrivere implementazioni personalizzate per sovrascriverli.

Il `get_model` metodo crea un `CustomProviderBuilder` (vedi il passaggio seguente) e chiama il `construct_chat_model` metodo che costruisce il modello di chat utilizzando i passaggi del generatore. Questo metodo funge da Director nel pattern del builder.

c. Copialo `clients/builders/bedrock_builder.py` e rinominalo in `custom_provider_builder.py` e la classe al suo interno in `CustomProviderBuilder` that inherits LLMBuilder (). `llm_builder.py` È possibile utilizzare i metodi forniti da LLMBuilder o scrivere implementazioni personalizzate per sovrascriverli. I passaggi del builder vengono richiamati in sequenza all'interno del `construct_chat_model` metodo del client, ad esempio`set_model_defaults`, `set_knowledge_base` e. `set_conversation_memory`

   Il `set_llm_model` metodo creerebbe il modello LLM effettivo utilizzando tutti i valori impostati utilizzando i metodi chiamati in precedenza. In particolare, è possibile creare un LLM RAG (`CustomProviderRetrievalLLM`) o non RAG (`CustomProviderLLM`), in base a quanto recuperato dalla configurazione LLM in DynamoDB. `rag_enabled variable`

   Questa configurazione viene recuperata nel metodo della classe. `retrieve_use_case_config` `LLMChatClient`

d. Implementate `CustomProviderRetrievalLLM` l'implementazione `CustomProviderLLM` or nella `llm_models` sottocartella a seconda che abbiate bisogno di un caso d'uso RAG o non RAG. La maggior parte delle funzionalità per implementare questi modelli sono fornite rispettivamente nelle rispettive `RetrievalLLM` classi, per casi d'uso diversi da RAG `BaseLangChainModel` e RAG.

   È possibile copiare il `llm_models/bedrock.py` file e apportare le modifiche necessarie per chiamare il LangChain modello che fa riferimento al provider personalizzato. Ad esempio, Amazon Bedrock utilizza una `ChatBedrock` classe per creare un modello di chat utilizzando LangChain.

   Il metodo generate genera la risposta LLM utilizzando le catene LangChain LCEL.

   È inoltre possibile utilizzare il `get_clean_model_params` metodo per disinfettare i parametri del modello in base ai requisiti LangChain del modello.

## Espansione degli strumenti Strands supportati

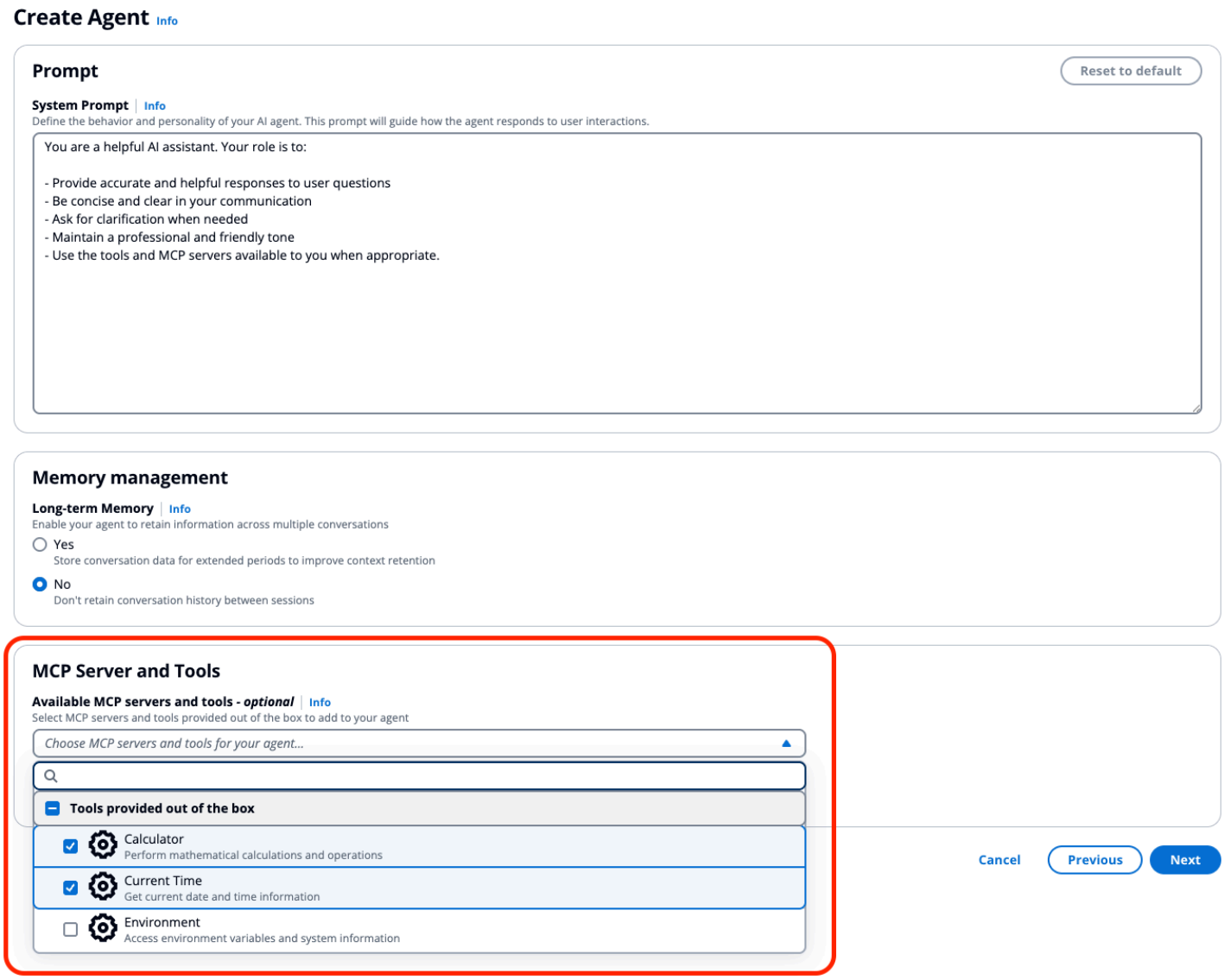
La soluzione consente di creare e distribuire server MCP, agenti AI e flussi di lavoro multiagente. Nell'ambito dell'esperienza Agent Builder, puoi collegare server MCP per offrire ai tuoi agenti

funzionalità aggiuntive. Oltre ai server MCP, è possibile sfruttare gli strumenti integrati forniti da Strands (il framework sottostante utilizzato dalla soluzione).

Pronta all'uso, la soluzione viene preconfigurata con i seguenti strumenti Strands:

- Ora corrente (abilitata per impostazione predefinita)
- Calcolatrice (abilitata per impostazione predefinita)
- Ambiente

Selezione del server e degli strumenti MCP nella procedura guidata di Agent Builder che mostra gli strumenti Strands integrati



Per ampliare i tuoi agenti con strumenti Strands aggiuntivi, segui la procedura in quattro fasi descritta in questa sezione.

## Fase 1: Trova lo strumento Strands

Sfoglia gli strumenti Strands disponibili per identificare lo strumento che desideri utilizzare. Ogni strumento ha funzionalità e requisiti di configurazione specifici.

Ad esempio, per aggiungere funzionalità di recupero della Knowledge Base di Amazon Bedrock, è necessario utilizzare lo strumento di recupero.

## Fase 2: Aggiornare il parametro SSM

Per rendere disponibile uno strumento nell'interfaccia utente di distribuzione di Agent Builder, aggiorna il parametro AWS Systems Manager Parameter Store che definisce quali strumenti Strands sono supportati.

1. Accedi all'AWS Systems Manager Parameter Store nel tuo account AWS.
2. Individua il parametro: `/gaab/<stack-name>/strands-tools`
3. Aggiungete la configurazione dello strumento alla fine dell'elenco esistente utilizzando la seguente struttura JSON:

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

| Campo | Description |
| --- | --- |
| name | Nome visualizzato nell'interfaccia utente di Agent Builder |
| description | Breve descrizione della funzionalità dello strumento |

| Campo | Description |
| --- | --- |
| value | Il nome esatto dello strumento come definito nel pacchetto Strands tools |
| category | Categoria organizzativa per il raggruppamento degli strumenti nell'interfaccia utente |
| è l'impostazione predefinita | Se lo strumento deve essere abilitato per impostazione predefinita per i nuovi agenti |

## Fase 3: Configurare le variabili di ambiente

Molti strumenti Strands richiedono variabili di ambiente per la configurazione. È possibile impostare queste variabili in due modi:

Opzione 1: configurazione diretta su AgentCore Runtime

Aggiorna l'agente distribuito direttamente su Amazon Bedrock AgentCore Runtime con le variabili di ambiente richieste.

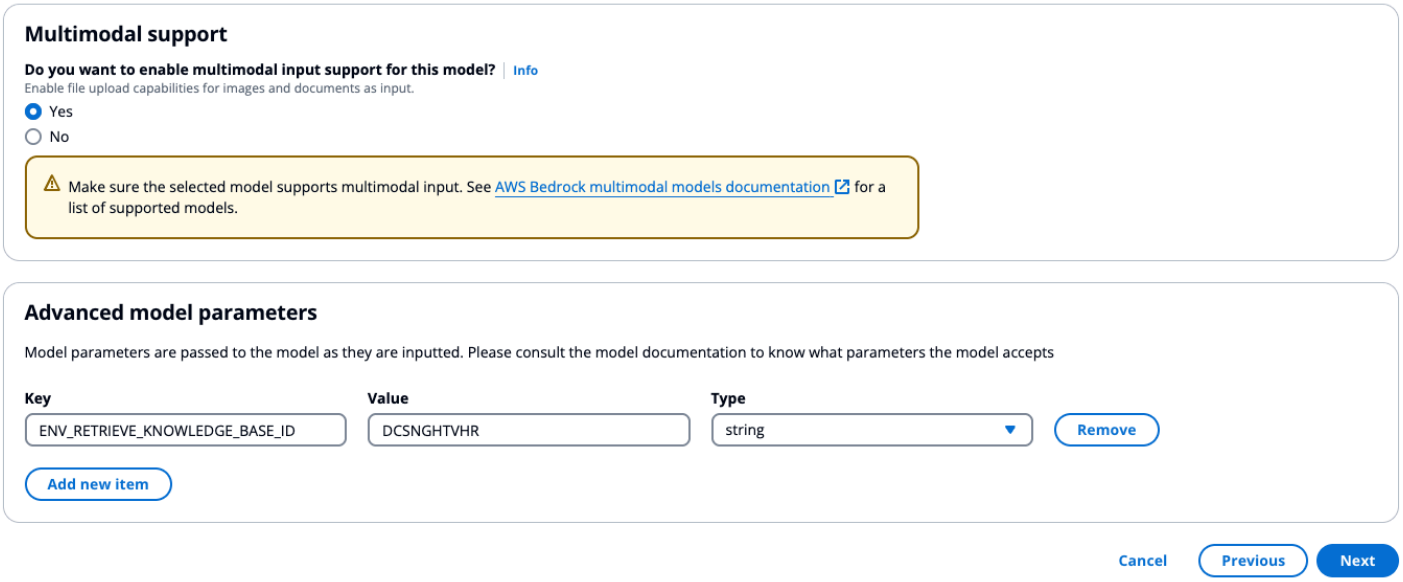
Opzione 2: parametri del modello nella procedura guidata di distribuzione

Aggiungere variabili di ambiente durante la fase di selezione del modello nella procedura guidata di Agent Builder utilizzando la sezione Parametri del modello. Le variabili di ambiente che seguono la convenzione di denominazione `ENV_<ALL_CAPS_TOOL_NAME>_<env_variable_name>` verranno caricate automaticamente in fase di esecuzione nell'ambiente di esecuzione dell'agente come. `<env_variable_name>`

Esempio:

- `ENV_RETRIEVE_KNOWLEDGE_BASE_ID`diventa `KNOWLEDGE_BASE_ID`
- `ENV_RETRIEVE_MIN_SCORE`diventa `MIN_SCORE`

Sezione dei parametri avanzati del modello che mostra la configurazione ENV_RETRIEVE_KNOWLEDGE_BASE_ID




Fate riferimento alla documentazione o al codice sorgente dello strumento specifico per identificare le variabili di ambiente richieste. Per lo strumento di recupero, puoi trovare le opzioni di configurazione nel codice sorgente.

## Fase 4: Aggiungere le autorizzazioni IAM

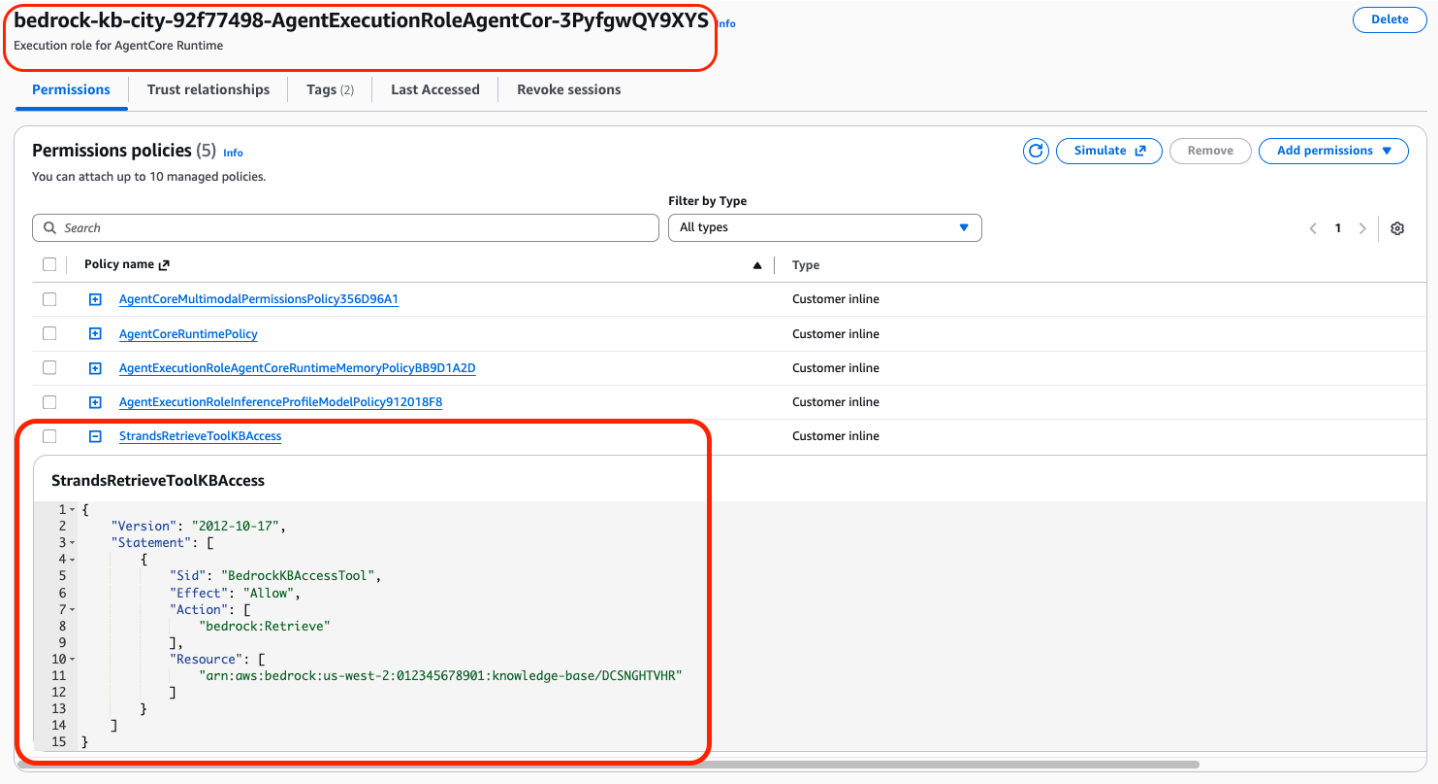
Aggiungi manualmente le autorizzazioni IAM necessarie al tuo ruolo AgentCore di esecuzione Runtime per consentire all'agente di utilizzare lo strumento.

Ad esempio, per utilizzare lo strumento di recupero con Amazon Bedrock Knowledge Bases:

1. Accedi alla console IAM nel tuo account AWS.
2. Individua il ruolo AgentCore di esecuzione in Runtime per il tuo agente.
3. Aggiungi la seguente autorizzazione:

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

Console IAM che mostra la StrandsRetrieveTool KBAccess policy associata al ruolo AgentCore di esecuzione Runtime

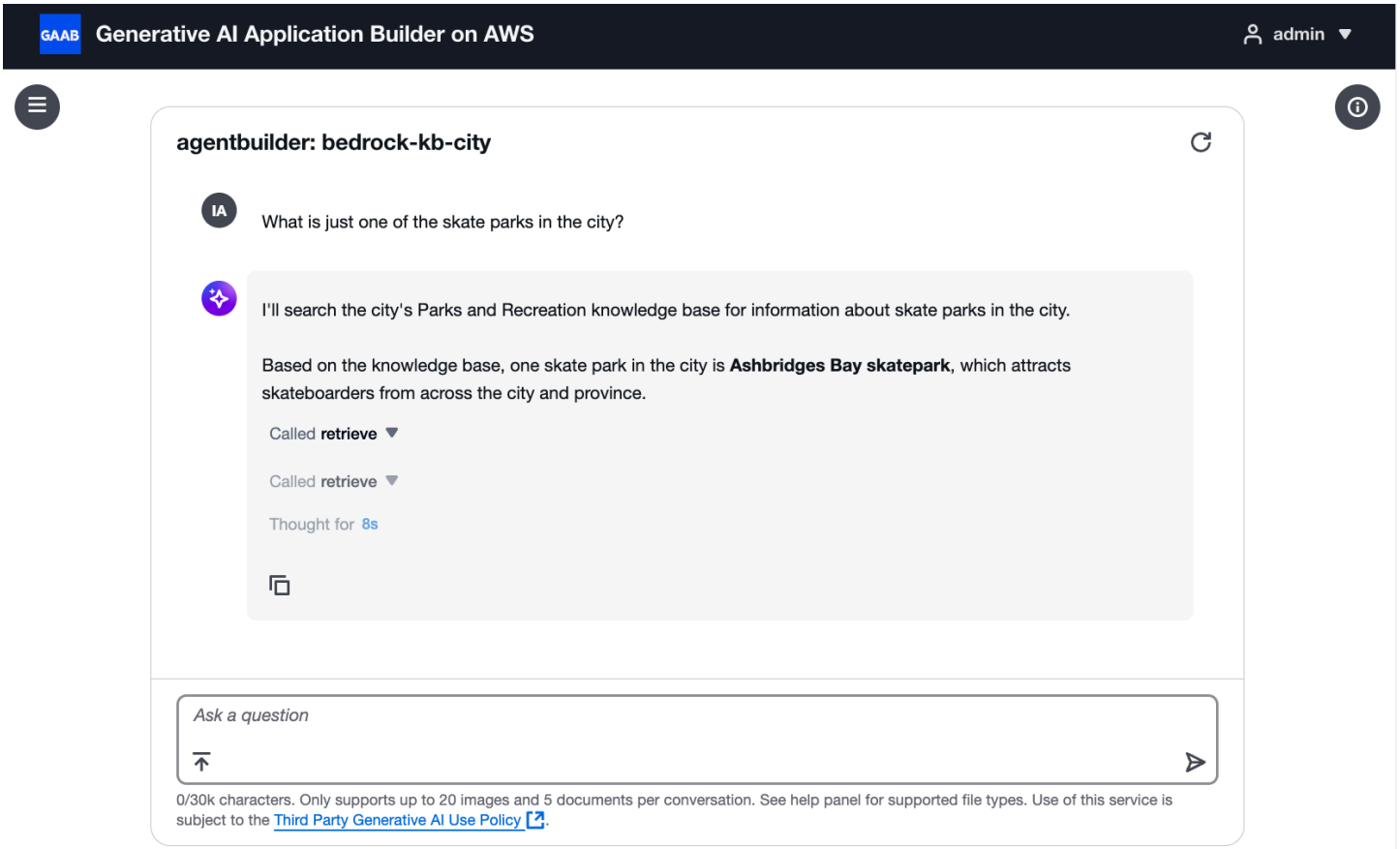



Le autorizzazioni specifiche richieste varieranno in base allo strumento. Consulta la documentazione dello strumento e la documentazione del servizio AWS per determinare le autorizzazioni IAM appropriate.

## Fase 5: testare l'agente

Dopo aver completato i passaggi di configurazione, testate l'agente per verificare che lo strumento funzioni correttamente. Dovresti vedere le chiamate agli strumenti nei registri di esecuzione e nelle risposte dell'agente.

Agente che utilizza con successo lo strumento di recupero per rispondere a una domanda sugli skate park




> **Note**
>
> Per un elenco completo degli strumenti Strands disponibili e delle relative funzionalità, consultate la documentazione degli Strands Community Tools.

## Ampliamento delle basi di conoscenza e dei tipi di memoria di conversazione supportati

Per aggiungere le tue implementazioni della memoria di conversazione o della knowledge base, aggiungi le implementazioni richieste `shared` nella cartella, quindi modifica le impostazioni di fabbrica e le enumerazioni appropriate per creare un'istanza di queste classi.

Quando fornite la configurazione LLM, che è memorizzata all'interno dell'archivio dei parametri, verranno create la memoria di conversazione e la knowledge base appropriate per il vostro LLM. Ad esempio, quando `ConversationMemoryType` viene specificato come DynamoDB, viene creata un'istanza `DynamoDBChatMessageHistory` di (disponibile `shared_components/memory/ddb_enhanced_message_history.py` all'interno). Quando `KnowledgeBaseType` viene

specificato come Amazon Kendra, viene creata un'istanza `KendraKnowledgeBase` di (disponibile `shared_components/knowledge/kendra_knowledge_base.py` all'interno).

## Creazione e implementazione delle modifiche al codice

Costruisci il programma con il `npm run build` comando. Una volta risolti gli errori, esegui `cdk synth` per generare i file modello e tutte le risorse Lambda.

1. Puoi utilizzare lo `0/stage-assets.sh` script per inserire manualmente le risorse generate nello staging bucket del tuo account.
2. Utilizzate il seguente comando per distribuire o aggiornare la piattaforma:

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

   Insieme al AdminUserEmailparametro devono essere forniti anche eventuali CloudFormation parametri AWS aggiuntivi.

# Guida alla personalizzazione

## Gestione del pool di utenti di Cognito

Quando viene distribuita la dashboard Deployment, viene creato un pool di utenti Amazon Cognito insieme a un utente amministratore per fornire l'autenticazione per l'applicazione. Questo pool di utenti è condiviso nella dashboard di Deployment e in tutti i casi d'uso. All'utente amministratore creato durante l'implementazione della dashboard viene automaticamente concesso l'accesso a tutti i casi d'uso distribuiti utilizzando la dashboard. Questo meccanismo è fornito tramite gruppi di pool di utenti di Amazon Cognito.

Quando un caso d'uso viene distribuito dalla dashboard, se viene fornita un'e-mail, verrà creato un utente nel pool di utenti condiviso, insieme a un gruppo di utenti denominato per il caso d'uso specifico. L'utente appena creato viene quindi aggiunto al gruppo, concedendo all'utente l'accesso allo use case.

Se desideri aggiungere un utente aggiuntivo a un determinato caso d'uso, puoi farlo creando un utente nel pool di utenti di Cognito e aggiungendolo ai gruppi corrispondenti ai casi d'uso a cui desideri che l'utente abbia accesso. Per una step-by-step guida, consulta Creazione di un nuovo utente nella Console di gestione AWS.

Allo stesso modo, se desideri creare utenti amministratori aggiuntivi, devi creare un nuovo utente e aggiungerlo al gruppo di amministratori nel pool di utenti.

I nomi utente vengono creati prendendo la parte dell'e-mail fornita prima di e aggiungendo l'@UUID del caso d'uso generato (o `-admin` nel caso dell'utente amministratore).

Nella scheda Gruppi, puoi vedere che un gruppo di amministratori e un gruppo per ogni caso d'uso sono stati creati automaticamente utilizzando il nome dello use case (come fornito nella procedura guidata) e l'UUID del caso d'uso.

# Guida di riferimento alle API

Questa sezione fornisce i riferimenti API per la soluzione.

## Dashboard di implementazione

| REST API | Metodo HTTP | Funzionalità | Chiamanti autorizzati |
|---|---|---|---|
| `/deployments` | `GET` | Ottieni tutte le implementazioni. | Token JWT autentica to da Amazon Cognito |
| `/deployments` | `POST` | Crea una nuova distribuzione di use case. | Token JWT autentica to da Amazon Cognito |
| `/deployments/{useCaseId}` | `GET` | Ottiene i dettagli di distribuzione per una singola distribuzione. | Token JWT autentica to da Amazon Cognito |
| `/deployments/{useCaseId}` | `PATCH` | Aggiorna una determinata distribuz ione. | Token JWT autentica to da Amazon Cognito |
| `/deployments/{useCaseId}` | `DELETE` | Elimina una determina ta distribuzione. | Token JWT autentica to da Amazon Cognito |
| `/model-info/use-case-types` | `GET` | Ottiene i tipi di casi d'uso disponibili per la distribuzione | Token JWT autentica to da Amazon Cognito |

| REST API | Metodo HTTP | Funzionalità | Chiamanti autorizzati |
|---|---|---|---|
| `/model-info/{useCaseType}/providers` | `GET` | Ottiene i fornitori di modelli disponibili per il tipo di caso d'uso specificato | Token JWT autenticato da Amazon Cognito |
| `/model-info/{useCaseType}/{providerName}` | `GET` | Ottiene i IDs modelli disponibili per un determinato provider e tipo di caso d'uso | Token JWT autenticato da Amazon Cognito |
| `/model-info/{useCaseType}/{providerName}/{modelId}` | `GET` | Ottiene le informazioni sul modello specificato, inclusi i parametri predefiniti. | Token JWT autenticato da Amazon Cognito |

> **Note**
>
> I file OpenAPI e Swagger possono anche essere esportati da API Gateway per una più facile integrazione con l'API. Vedi Esportazione di un'API REST da API Gateway.

### Payload POST e PATCH

Di seguito è riportato un esempio di payload POST sull'`/deployments`endpoint, che creerà un nuovo caso d'uso.

```
{
 "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
 }
 }

}
```

Per gli aggiornamenti, la struttura è la stessa di cui sopra con alcune avvertenze:

- Il nome del caso d'uso non può essere modificato
- Uno use case può modificare i gruppi di sicurezza e le sottoreti solo dopo essere stato distribuito in un VPC. Il VPC stesso non può essere modificato.
- Se un indice Kendra è stato creato per te come knowledge base, non puoi modificare la configurazione di quell'indice (ad esempio,,) KendraIndexName QueryCapacityUnits

## Caso d'uso condiviso APIs

I seguenti endpoint dell'API REST sono disponibili per i casi d'uso di Text e Bedrock Agent:

| REST API | Metodo HTTP | Funzionalità | Chiamanti autorizzati |
|---|---|---|---|
| `/details/{useCaseConfigKey}` | `GET` | Ottiene i dettagli di configurazione per un caso d'uso specifico. | Token JWT autenticato da Amazon Cognito |

| WebSocket API | Funzionalità | Chiamanti autorizzati |
|---|---|---|
| `/$connect` | Avvia la WebSocket connessione e autentica l'utente. | Token JWT autenticato da Amazon Cognito |
| `/$disconnect` | Endpoint chiamato quando una WebSocket connessione è stata interrotta. | Token JWT autenticato da Amazon Cognito |

### Usa l'API Case Details

L'endpoint dell'API details recupera informazioni su un caso d'uso specifico:

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

Questo endpoint restituisce i dettagli di configurazione per un caso d'uso specifico, inclusi i parametri del modello, le impostazioni della knowledge base e altre informazioni sulla distribuzione. Richiede un token JWT autenticato da Amazon Cognito per l'autorizzazione.

## Caso di utilizzo del testo

| WebSocket API | Funzionalità | Chiamanti autorizzati |
|---|---|---|
| `/sendMessage` | Invia il messaggio di chat dell'utente a WebSocket per l'elaborazione con l'esperienza LLM configurata. | Token JWT autenticato da Amazon Cognito |

| REST API | Metodo HTTP | Funzionalità | Chiamanti autorizzati |
|---|---|---|---|
| `/feedback/{useCaseId}` | POST | Invia il feedback degli utenti per un caso d'uso specifico. | Token JWT autenticato da Amazon Cognito |

### Payload SendMessage

Se ti stai integrando direttamente con l'`/sendMessage`API, devi rispettare i seguenti formati di payload di richiesta e risposta.

### Richiedi Payload

```
{
 "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
}
```

| Nome parametro | Tipo | Description |
| --- | --- | --- |
| action | `String` | Attualmente supportiamo solo l'azione «SendMessage» su WebSocket |
| domanda | `String` | L'input dell'utente da inviare al LLM |
| ID di conversazione | `String` | Un UUID che identifica la conversazione. Se non viene fornito, verrà creata una nuova conversazione, con il ConversationID restituito nella risposta. Tutti i messaggi successivi di quella conversaz ione (in cui desideri che vengano mantenuti), devono history/context fornire lì il ConversationID. |
| Modello di prompt | `String` [Opzionale] | Sostituisce il modello di prompt per questo messaggio . Se vuoto o non fornito, verrà utilizzato per impostazione predefinita il prompt impostato al momento della distribuz ione. Deve avere i segnapost o appropriati specificati per la configurazione data (ad esempio {history} e {input} per le implementazioni AI non RAG Sagemaker, con l'aggiunt |

| Nome parametro | Tipo | Description |
| --- | --- | --- |
| | | a di {context} se si utilizza RAG per tutte le implement azioni. |
| AuthToken | `String` [Opzionale] | AccessToken ottenuto dal flusso di autenticazione cognito. Ciò è necessari o quando si richiama un endpoint websocket di chat configurato per RAG con Role Based Access Control (RBAC). L'elenco dei claim cognito:groups in questo token JWT viene utilizzat o per controllare l'accesso ai documenti nell'indice Kendra. Questo parametro non è richiesto per casi d'uso diversi da RAG. Inoltre, non è necessario per i casi d'uso RAG con RBAC disabilitato. |

Payload di risposta

Risposta alla domanda

L' WebSocket API risponderà con 1 (se lo streaming è disabilitato) o molti (se lo streaming è abilitato) oggetti JSON strutturati come segue per ogni query.

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

| Nome parametro | Tipo | Description |
| --- | --- | --- |
| dati | `String` | Una parte della risposta del LLM, se lo streaming è abilitato, o l'intera risposta. Se si utilizza lo streaming, verrà inviata una risposta di questo formato con il contenuto dei dati END_CONVERSATION per indicare la fine della risposta a una singola domanda. |
| ConversationID | `String` | L'ID della conversazione a cui appartiene questa risposta SourceDocument. |

Risposta del documento di origine

Se avete configurato lo use case RAG per restituire documenti di origine, riceverete anche il seguente payload alla fine di ogni risposta per ogni documento sorgente utilizzato per creare la risposta.

```
{
 "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"
}
```

| Nome parametro | Tipo | Description |
| --- | --- | --- |
| estratto | `String` | Un estratto dal documento sorgente. |
| posizione | `String` | Ubicazione del documento di origine. Ciò dipenderà dalle fonti di dati utilizzate e dal tipo di knowledge base, ma potrebbero essere cose come s3 URIs o siti Web. |
| punteggio | `Number` \| `String` | La certezza che il documento corrisponda alla domanda posta. Questo sarà un float da 0 a 1 per Bedrock e una stringa (ad esempio HIGH, LOW, ecc.) per Kendra. |
| titolo_documento | `String` | Titolo del documento sorgente restituito. Disponibile solo quando usi Kendra. |
| document_id | `String` | ID del documento sorgente restituito. Disponibile solo quando usi Kendra. |
| attributi_aggiuntivi | `String` | Questo campo conterrà tutti gli attributi aggiuntiv i del documento, così come personalizzati nella Knowledge Base al momento dell'inserimento. |
| ConversationID | `String` | L'ID della conversazione a cui appartiene questa risposta SourceDocument. |

Payload dell'API di feedback

Di seguito è riportato un esempio di payload POST sull'`/feedback/{useCaseId}`endpoint, che invierà il feedback degli utenti per un caso d'uso specifico:

```
{
  "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"
  ]
}
```

## Caso d'uso di Bedrock Agent

| WebSocket API | Funzionalità | Chiamanti autorizzati |
| --- | --- | --- |
| `/invokeAgent` | Invia il messaggio dell'utente a WebSocket per l'elaborazione con l'agente configurato. | Token JWT autenticato da Amazon Cognito |

### Carichi utili InvokeAgent

Se ti stai integrando direttamente con`/invokeAgent API`, devi rispettare i seguenti formati di payload di richiesta e risposta.

Payload della richiesta

```
{
"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
}
```

| Nome del parametro | Tipo | Description |
|---|---|---|
| action | `String` | Supportiamo solo l'`invokeAgent` azione su. WebSocket |
| Testo di input | `String` | L'input dell'utente da inviare al LLM. |
| ID di conversazione | `String[Optional]` | Un UUID che identifica in modo univoco la conversaz ione. Se non fornisci questo valore, la soluzione crea una nuova conversazione e restituisce il ConversationID nella risposta. Tutti i messaggi successivi di quella conversaz ione (in cui desideri conservar e la cronologia e il contesto) forniscono lì il ConversationID. |
| AuthToken | `String[Optional]` | AccessToken è stato ottenuto dal flusso di autenticazione di Amazon Cognito. Questo parametro non è obbligatorio. Se lo fornisci, il token JWT verrà convalidato. Questo aiuta a semplificare l'estensi one di questa soluzione. |

Carichi utili di risposta

### Risposta alla domanda

L' WebSocket API risponderà con uno (se lo streaming è disabilitato) o più (se lo streaming è abilitato) oggetti JSON strutturati come segue per ogni query.

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

| Nome del parametro | Tipo | Description |
|---|---|---|
| dati | `String` | La risposta della chiamata dell'agente. |
| ConversationID | `String` | L'ID della conversazione. |

# Riferimento

Questa sezione include informazioni sulla raccolta dei dati per questa soluzione, riferimenti a risorse correlate e un elenco di costruttori che hanno contribuito a questa soluzione.

## Provider LLM supportati

La soluzione può integrarsi con i seguenti provider LLM:

1. Amazon Bedrock
   - Documentazione: https://aws.amazon.com/bedrock/
   - Modelli supportati:
     - Amazon
       - Nova Lite
       - Nova Micro
       - Nova Pro
     - AI21 Laboratori
       - Jamba 1.5 Mini
       - Jamba 1.5 Large
     - Anthropic
       - Haiku Claude v3
       - Claude v3.5 Sonetto
       - Claude v3.7 Sonnet (attraverso l'uso di profili di inferenza)
     - Cohere
       - Comando R
       - Comando R
     - Deepseek
       - Deepseek-R1 (attraverso l'uso di profili di inferenza)
     - Meta
       - Llama 3
       - Llama 3.2 (attraverso l'uso di profili di inferenza)
     - Mistral AI

        - Mistral 7B Instruct
        - Mistral 8x7B Istruct
    - Inferenza tra regioni
        - Possibilità di utilizzare i profili di inferenza definiti nella stessa regione del dashboard di distribuzione
2. Amazon SageMaker AI
    - Documentazione: https://aws.amazon.com/sagemaker/
    - Modelli supportati: modelli da testo a testo

Per i parametri più recenti del modello, le best practice e gli usi consigliati, consulta la documentazione dei fornitori di modelli.

## Raccolta dei dati

Questa soluzione invia metriche operative ad AWS (i «Dati») sull'utilizzo di questa soluzione. Utilizziamo questi dati per comprendere meglio come i clienti utilizzano questa soluzione e i servizi e i prodotti correlati. La raccolta di questi dati da parte di AWS è soggetta all'Informativa sulla privacy di AWS.

## Collaboratori

- Tarek Abdunabi
- Majd Arbash
- George Bearden
- Mukit Bin Momin
- Michael Connor
- Johnny Duval
- Nihit Kasabwala
- Ahern Knox
- Simon Krol
- Michael Lin
- Tim Mekari

- Ibrahim Mohamed
- Omar Radwan Mohsen
- James Nixon
- Dekshita Ravikumar
- Jae Shim
- Ajay Swamy
- Mohamed Taha
- Reet Takkar
- Dimitri Tchikatilov
- Jason Ghirlanda
- Kamyar Ziabari

# Revisioni

Data di pubblicazione: ottobre 2023 (ultimo aggiornamento: gennaio 2025)

Controlla il file Changelog.md nel GitHub repository per vedere tutte le modifiche e gli aggiornamenti importanti del software. Il changelog fornisce una chiara registrazione dei miglioramenti e delle correzioni per ogni versione.

# Note

I clienti sono responsabili della propria valutazione indipendente delle informazioni contenute nel presente documento. Questo documento: (a) è solo a scopo informativo, (b) rappresenta le attuali offerte e pratiche di prodotti AWS, che sono soggette a modifiche senza preavviso, e (c) non crea alcun impegno o garanzia da parte di AWS e delle sue affiliate, fornitori o licenzianti. I prodotti o i servizi AWS sono forniti «così come sono» senza garanzie, dichiarazioni o condizioni di alcun tipo, esplicite o implicite. Le responsabilità e gli obblighi di AWS nei confronti dei propri clienti sono controllati da accordi AWS e questo documento non fa parte né modifica alcun accordo tra AWS e i suoi clienti.

Generative AI Application Builder su AWS è concesso in licenza secondo i termini della versione 2.0 della licenza Apache.

**Important**

Generative AI Application Builder on AWS ti consente di creare e distribuire applicazioni di intelligenza artificiale generativa su AWS utilizzando il modello di intelligenza artificiale generativa di tua scelta, inclusi modelli di intelligenza artificiale generativa di terze parti che puoi scegliere di utilizzare e su cui AWS non possiede o su cui non ha alcun controllo («Modelli di IA generativa di terze parti»).
L'utilizzo dei Modelli di IA generativa di terze parti è regolato dai termini forniti all'utente dai fornitori di modelli di intelligenza artificiale generativa di terze parti al momento dell'acquisizione della licenza per utilizzarli (ad esempio, i termini di servizio, il contratto di licenza, la politica d'uso accettabile e l'informativa sulla privacy).
L'utente è responsabile di garantire che l'utilizzo dei Modelli di IA generativa di terze parti sia conforme ai termini che li disciplinano e a tutte le leggi, norme, regolamenti, politiche o standard applicabili all'utente.
L'utente è inoltre responsabile della valutazione indipendente dei modelli di intelligenza artificiale generativa di terze parti che utilizza, compresi i relativi risultati e il modo in cui i fornitori di modelli di intelligenza artificiale generativa di terze parti utilizzano i dati che potrebbero essere trasmessi loro in base all'implementazione da parte dell'utente. AWS non rilascia alcuna dichiarazione o garanzia in merito ai modelli di intelligenza artificiale generativa di terze parti, che sono «contenuti di terze parti» ai sensi del contratto con AWS. Generative AI Application Builder su AWS viene offerto come «Contenuto AWS» ai sensi del contratto con AWS.

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.