Guía de implementación

Creador de aplicaciones de IA generativa en AWS

Creador de aplicaciones de IA generativa en AWS: Guía de implementación

Table of Contents

Esta solución facilita el desarrollo, la experimentación rápida y el despliegue de aplicaciones de inteligencia artificial (IA) generativa

El generador de aplicaciones de IA generativa en AWS facilita el desarrollo, la experimentación rápida y la implementación de aplicaciones de inteligencia artificial (IA) generativa sin necesidad de una amplia experiencia en IA. Esta solución de AWS acelera el desarrollo y agiliza la experimentación al ayudarlo a:

- Ingiera los datos y documentos específicos de su empresa
- Evalúe y compare el rendimiento de modelos lingüísticos de gran tamaño () LLMs
- Ejecute tareas y flujos de trabajo de varios pasos con agentes de IA
- Cree aplicaciones ampliables con rapidez e impleméntelas con una arquitectura de nivel empresarial

Generative AI Application Builder en AWS incluye integraciones con:

- LLMs disponible en Amazon Bedrock
- LLMs que ha implementado en Amazon SageMaker AI
- Bases de conocimiento de Amazon Bedrock para la generación aumentada de recuperación (RAG)
- Amazon Bedrock Guardrails implementará salvaguardas y reducirá las alucinaciones
- Amazon Bedrock Agents creará flujos de trabajo de agentes que puedan llevar a cabo la orquestación y finalización de tareas
- Amazon Bedrock AgentCore construirá, implementará y administrará agentes de IA listos para la producción con soporte de tiempo de ejecución extendido
- Modele servidores de protocolo de contexto (MCP) para la integración de herramientas y datos empresariales

Además, esta solución permite realizar conexiones con el modelo que elija mediante LangChain conectores. Estos conectores están disponibles en una función de AWS Lambda que se implementa con la solución. Puede empezar con el asistente de implementación sin código para crear

aplicaciones de IA generativas para la búsqueda conversacional, los chatbots generados por la IA, la generación de texto y el resumen de texto.

Esta guía de implementación proporciona una descripción general de la solución Generative AI Application Builder en AWS, su arquitectura y componentes de referencia, las consideraciones para planificar la implementación y los pasos de configuración para implementar la solución en la nube de Amazon Web Services (AWS).

Esta guía está destinada a arquitectos de soluciones, responsables de la toma de decisiones empresariales, DevOps ingenieros, científicos de datos y profesionales de la nube que desean implementar Generative AI Application Builder en AWS en su entorno.

Utilice esta tabla de navegación para encontrar rápidamente las respuestas a estas preguntas:

Si quiere…	Lea…
Conocer el costo de ejecutar esta solución. El costo estimado de ejecutar esta solución varía en función de los componentes que implemente y del número de consultas. El costo de ejecutar el panel de implement ación con los parámetros predeterminados y 100 usuarios activos en la región de EE. UU. del Este (Virginia del Norte) durante un mes es de aproximadamente 20,12 USD al mes. El coste de un caso de uso de texto implement ado sin RAG para un usuario empresarial que realice 100 consultas al día con el LLM es de aproximadamente 12,39 USD al mes. El coste de un caso de uso habilitado para RAG con un índice de Amazon Kendra que soporte 8000 interacciones al día es de aproximadamente 204,26 USD al mes, más el coste de la base de conocimientos.	Costo

Si quiere…	Lea…
Comprender las consideraciones de seguridad de esta solución.	Seguridad
Saber cómo planificar las cuotas de esta solución.	Cuotas
Conozca qué regiones de AWS admiten esta solución.	Regiones de AWS admitidas
Consulte o descargue la CloudFormation plantilla de AWS incluida en esta solución para implementar automáticamente los recursos de infraestructura (la «pila») de esta solución.	CloudFormation Plantilla de AWS
Acceder al código fuente y, opcionalm ente, utilizar AWS Cloud Development Kit (AWS CDK) para implementar la solución.	GitHub repositorio

Características y ventajas

La solución Generative AI Application Builder en AWS ofrece las siguientes funciones:

Experimentación rápida

Esta solución permite a los usuarios experimentar rápidamente al eliminar el trabajo pesado que supone implementar varias instancias con diferentes configuraciones y comparar los resultados y el rendimiento. Experimente con múltiples configuraciones de varios LLMs parámetros: ingeniería rápida, bases de conocimiento empresarial, barreras, agentes de IA y otros parámetros.

Elección y configurabilidad

Con conectores preintegrados para una variedad de modelos LLMs, como los disponibles en Amazon Bedrock, esta solución le brinda la flexibilidad de implementar el modelo que prefiera, así como los principales servicios de AWS y FM que prefiera. También puede permitir que los agentes de Amazon Bedrock realicen diversas tareas y flujos de trabajo.

Agent Builder

Cree e implemente agentes de IA listos para la producción con una gestión completa del ciclo de vida. Configure las indicaciones del sistema, integre los servidores del Model Context Protocol (MCP) para el acceso a los datos y las herramientas empresariales, y habilite las funciones de memoria para retener el contexto en todas las conversaciones. Los agentes se despliegan en Amazon Bedrock AgentCore con soporte de tiempo de ejecución ampliado y respuestas de streaming en tiempo real.

Creador de flujos de trabajo

Organice varios agentes de Agent Builder en flujos de trabajo complejos mediante la delegación jerárquica. Cree un agente supervisor que seleccione y coordine de forma autónoma a los agentes especializados de Agent Builder para que se encarguen de tareas de varios pasos. Configure las descripciones de los agentes, las estrategias de delegación y la memoria a nivel de flujo de trabajo mientras reutiliza las implementaciones de Agent Builder existentes.

Listo para la producción

Creada con los principios de diseño de Well-Architected de AWS, esta solución ofrece seguridad y escalabilidad de nivel empresarial con alta disponibilidad y baja latencia, lo que garantiza una integración perfecta en sus aplicaciones con altos estándares de rendimiento.

Arquitectura modular extensible

Amplíe la funcionalidad de esta solución integrando sus proyectos existentes o conectando servicios de AWS adicionales de forma nativa. Como se trata de una aplicación de código abierto, puede utilizar la capa de LangChain orquestación incluida o las funciones Lambda para conectarse con los servicios que prefiera.

Integración con Service Catalog AppRegistry y Application Manager, una funcionalidad de AWS Systems Manager

Esta solución incluye un AppRegistry recurso del catálogo de servicios para registrar la CloudFormation plantilla de la solución y sus recursos subyacentes como una aplicación tanto en AWS Service Catalog AppRegistry como en AWS Systems Manager Application Manager. Con esta integración, puede administrar de forma centralizada los recursos de la solución.

Caso de uso entre Agent Builder y Bedrock Agent

Esta solución ofrece dos enfoques distintos para trabajar con agentes de IA, cada uno adecuado para diferentes casos de uso y requisitos:

Característica	Caso de uso de Bedrock Agent	Agent Builder
Finalidad	Invoque los agentes Amazon Bedrock desplegados previamente	Cree, implemente y gestione agentes personalizados
Configuración	Solo ID de agente e ID de alias	Configuración completa del agente: indicaciones del sistema, modelos, servidores MCP, memoria
Implementación	Capa de invocación sencilla	Ciclo de vida completo del agente en tiempo de ejecución AgentCore
Tiempo de ejecución	Servicio Amazon Bedrock Agents	Amazon Bedrock AgentCore con el SDK de Strands
Integración de herramientas	Configurado en la consola Bedrock Agents	Modele servidores de protocolo de contexto (MCP) y herramientas Strands integradas
Memoria	Administrado por Bedrock Agents (hasta 30 días)	AgentCore Memoria con retención configurable a corto y largo plazo
Personalización	Limitado a la configuración del agente previamente implementada	Control total sobre las indicaciones, los modelos, las herramientas y el comportamiento
Lo mejor para	Despliegue rápido de los agentes existentes	Implementaciones personalizadas de desarrollo y producción de agentes

> **Note**
>
> Ambas opciones admiten la transmisión en tiempo real, el historial de conversaciones y la seguridad de nivel empresarial.

Creador de flujos de trabajo

Workflow Builder permite la organización de varios agentes mediante la creación de un agente supervisor que delega el trabajo en agentes especializados de Agent Builder. Cada flujo de trabajo consta de:

- Agente supervisor: el agente de punto de entrada que recibe las solicitudes de los usuarios y coordina a los agentes especializados
- Agentes especializados: casos de uso de Agent Builder en los que el supervisor puede delegar tareas
- Patrón de agentes como herramientas: el supervisor registra cada agente de Agent Builder como una herramienta y selecciona de forma autónoma qué agentes usar

Característica	Agent Builder	Creador de flujos de trabajo
Finalidad	Cree e implemente agentes personalizados únicos	Organice varios agentes de Agent Builder
Tipo de agente	Agente único con herramientas MCP	Agente supervisor y varios agentes de Agent Builder
Integración de herramientas	Servidores MCP y herramientas Strands	Agentes de Agent Builder registrados como herramientas
Delegación	Invocación directa de herramientas	Selección y delegación autónomas de agentes
Complejidad	Tareas con un solo agente	Flujos de trabajo de varios pasos y múltiples agentes
Reutilización de agentes	N/A	Reutiliza las implementaciones de Agent Builder existentes
Lo mejor para	Tareas enfocadas en un solo dominio	Flujos de trabajo complejos que requieren múltiples especializaciones

> **Note**
>
> - Los flujos de trabajo requieren al menos un caso de uso de Agent Builder como agente especializado
> - Todos los agentes especializados deben ser casos de uso de Agent Builder implementados en la GAAB

Casos de uso

Respuesta a preguntas sobre datos empresariales

LLMs y otros modelos básicos se han entrenado previamente en un gran corpus de datos, lo que les permite desempeñarse bien en muchas tareas de procesamiento del lenguaje natural (PNL). Sin embargo, la mayoría de los modelos básicos LLMs son estáticos y han sido entrenados previamente, lo que limita su capacidad para responder con precisión a preguntas sobre temas nuevos, especializados o patentados. Al utilizar el aprendizaje basado en indicaciones, puede aprovechar las potentes funciones de PNL y generación de texto de un LLM para ofrecer experiencias de cliente más enriquecedoras con los datos de su empresa.

Creación rápida de prototipos de IA generativa

La solución viene lista para usar e incluye varios proveedores de modelos y casos de uso. Con un asistente de implementación fácil de usar, los clientes pueden implementar casos de uso prediseñados para permitir la experimentación rápida de diferentes prototipos y cargas de trabajo de IA generativa.

Comparación y experimentación con varios LLM

LLMs funcionan de forma diferente y, dadas las necesidades específicas de su aplicación, es posible que un LLM se adapte mejor a su aplicación que otro. Esto puede deberse a motivos relacionados con el rendimiento, la precisión, el coste, la creatividad o muchos otros factores. Esta solución le permite implementar rápidamente varios casos de uso, lo que le permite experimentar y comparar diferentes configuraciones hasta encontrar la que satisfaga sus necesidades.

Conceptos y definiciones

En esta sección se describen los conceptos clave y se define la terminología específica de esta solución:

usuario administrador

En el contexto de esta guía, el usuario administrador es el responsable de administrar el contenido de la implementación. Este usuario tiene acceso a la interfaz de usuario del panel de implementación y es el principal responsable de organizar la experiencia del usuario empresarial. Este es nuestro principal cliente objetivo.

usuario empresarial

En el contexto de esta guía, el usuario empresarial representa a las personas para las que se ha implementado el caso de uso. Son los consumidores de la base de conocimientos y el cliente responsable de evaluarla y experimentar con ella. LLMs

Panel de despliegue

El panel de implementación es una interfaz web que sirve como consola de administración para que los usuarios administradores vean, administren y creen sus casos de uso. Este panel permite a los clientes experimentar, iterar y poner en producción diversas AI/ML cargas de trabajo de forma rápida y aprovecharlas. LLMs

DevOps usuario

En el contexto de esta guía, el DevOps usuario es el responsable de implementar la solución en la cuenta de AWS y de administrar la infraestructura, actualizar la solución, monitorear el rendimiento y mantener el estado general y el ciclo de vida de la solución.

caso de uso

Los casos de uso son aplicaciones aisladas de la solución general que se integran LLMs para ofrecer experiencias de cliente más enriquecedoras al permitir la adición de una interfaz de lenguaje natural a las aplicaciones nuevas o existentes. Los casos de uso se pueden implementar a través del panel de implementación o por sí solos.

> **Note**
>
> Para ver una referencia general de los términos de AWS, consulte el Glosario de AWS.

Información general de la arquitectura

En esta sección, se proporcionan diagramas de arquitectura de implementación de referencia para los componentes implementados con esta solución.

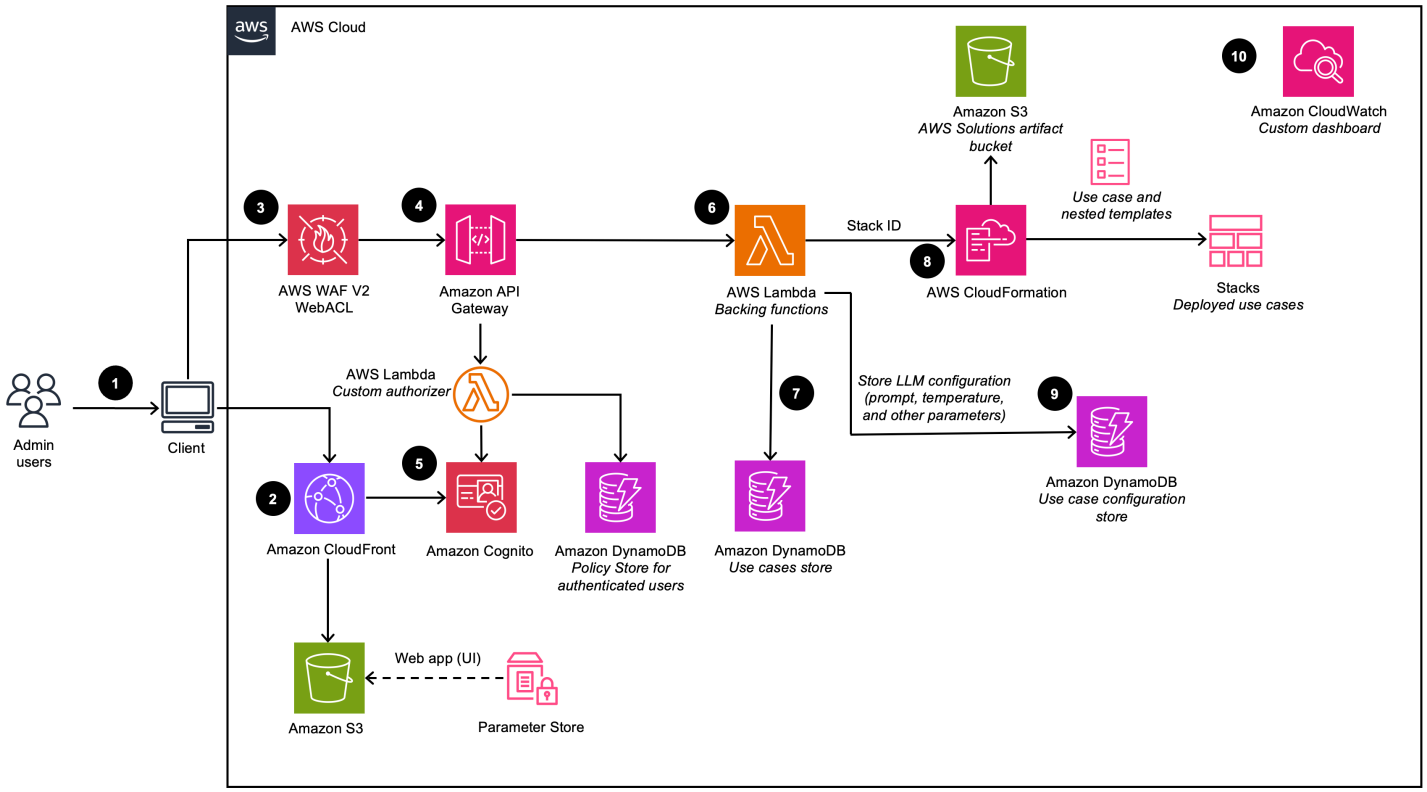
Diagramas de arquitectura

Para dar soporte a múltiples casos de uso y necesidades empresariales, esta solución ofrece seis CloudFormation plantillas de AWS:

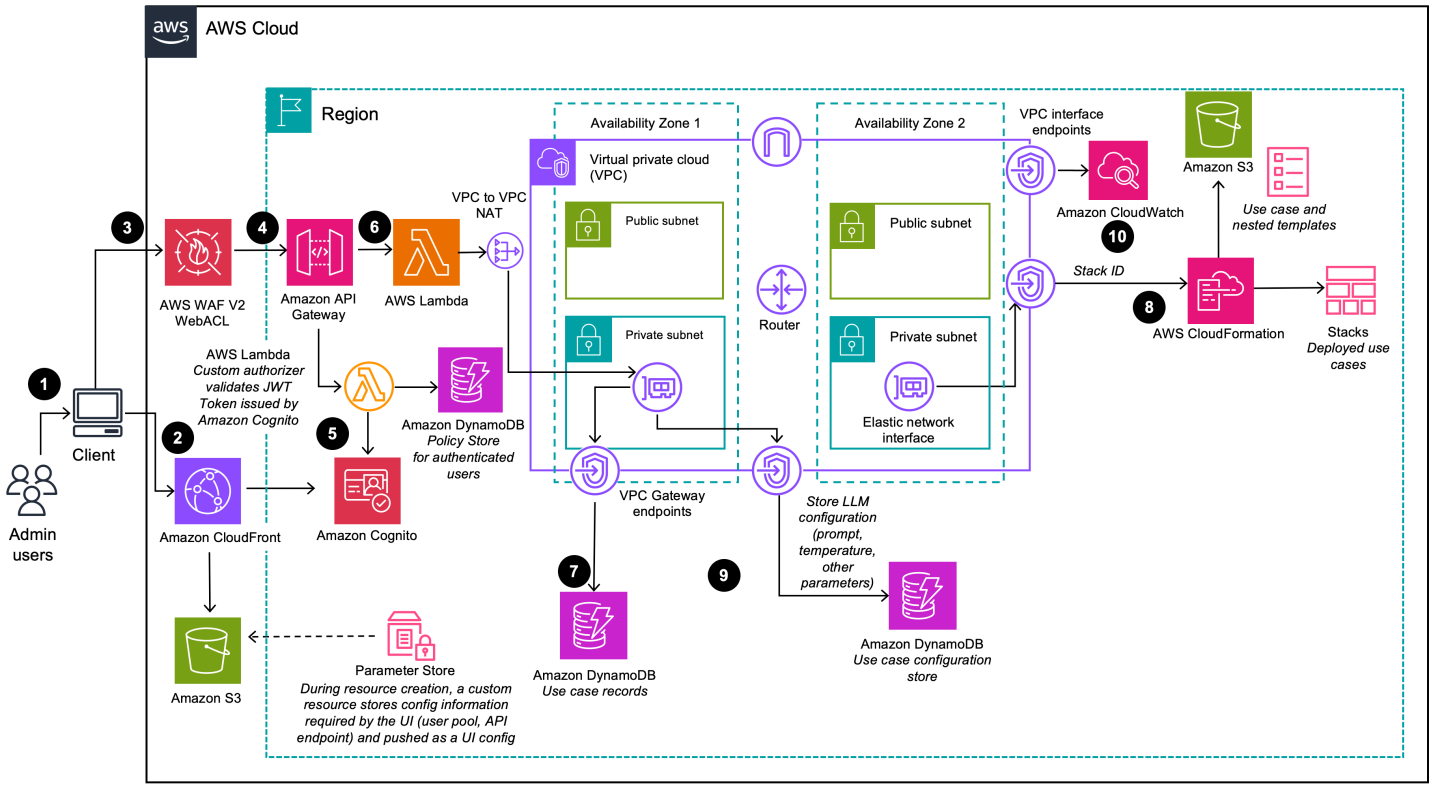
1. Panel de implementación: el panel de implementación es una interfaz web que sirve como consola de administración para que los usuarios administradores vean, administren y creen sus casos de uso. Este panel de control permite a los clientes experimentar, iterar y poner en producción diversas AI/ML cargas de trabajo con rapidez, aprovechando el potencial. LLMs
2. Caso de uso de texto: el caso de uso de texto permite a los usuarios disfrutar de una interfaz de lenguaje natural mediante la IA generativa. Este caso de uso se puede integrar en aplicaciones nuevas o existentes y se puede implementar a través del panel de implementación o de forma independiente a través de la URL proporcionada.
3. Caso de uso de Bedrock Agent: el caso de uso de Bedrock Agent permite utilizar los agentes de Bedrock existentes para completar tareas o automatizar flujos de trabajo repetidos.
4. Servidor MCP: el caso de uso del servidor MCP permite el despliegue y la administración de servidores del Model Context Protocol que proporcionan acceso estandarizado a herramientas y recursos a las aplicaciones de IA. Admite tanto los métodos de puerta de enlace para empaquetar las funciones Lambda existentes como los servidores MCP externos APIs, y los métodos de tiempo de ejecución para implementar servidores MCP en contenedores personalizados.
5. Agent Builder: Agent Builder permite la creación e implementación de agentes de IA listos para la producción en Amazon Bedrock AgentCore con un control total de la configuración, integración de servidores MCP y capacidades de administración de memoria.
6. Generador de flujos de trabajo: el generador de flujos de trabajo permite crear agentes supervisores que orquesten varios agentes de Agent Builder utilizando el patrón de delegación Agents as Tools para flujos de trabajo complejos con varios agentes.

Panel de despliegue

Representa la arquitectura del panel de implementación (cuando se implementa con la opción VPC deshabilitada)



Describe la arquitectura del panel de implementación (cuando se implementa con la opción VPC habilitada)



> **Note**
>
> Los CloudFormation recursos de AWS se crean a partir de componentes del AWS Cloud Development Kit (AWS CDK).

El flujo de proceso de alto nivel para los componentes de la solución implementados con la CloudFormation plantilla de AWS es el siguiente:

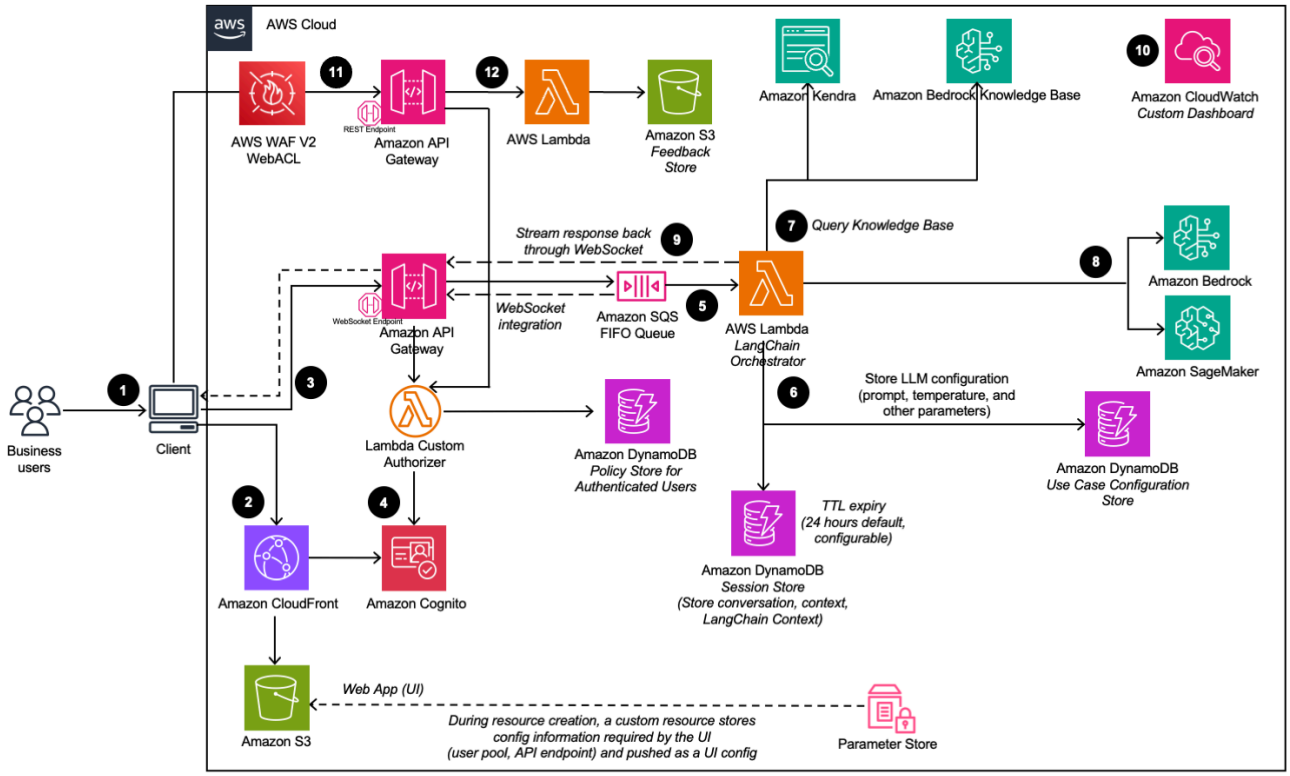
1. Los usuarios administradores inician sesión en la interfaz de usuario (UI) del Deployment Dashboard.
2. Amazon CloudFront ofrece la interfaz de usuario web, que se aloja en un bucket de Amazon Simple Storage Service (Amazon S3).
3. AWS WAF los protege APIs de los ataques. Esta solución configura un conjunto de reglas denominado lista de control de acceso a la web (ACL web) que permite, bloquea o cuenta las solicitudes web en función de reglas y condiciones de seguridad web configurables y definidas por el usuario.
4. La interfaz de usuario web utiliza un conjunto de REST APIs que se exponen mediante Amazon API Gateway.

5. Amazon Cognito autentica a los usuarios y respalda tanto la interfaz de usuario CloudFront web como la API Gateway.
6. AWS Lambda proporciona la lógica empresarial para los puntos de enlace REST. Esta función Lambda de respaldo administra y crea los recursos necesarios para realizar implementaciones de casos de uso mediante AWS. CloudFormation
7. Amazon DynamoDB almacena la lista de implementaciones.
8. Cuando el usuario administrador crea un nuevo caso de uso, la función Lambda de respaldo inicia un evento de creación de CloudFormation pilas para el caso de uso solicitado.
9. Todas las opciones de configuración de LLM proporcionadas por el usuario administrador en el asistente de implementación se guardan en DynamoDB. La implementación usa esta tabla de DynamoDB para configurar el LLM en tiempo de ejecución.
10. Con Amazon CloudWatch, esta solución recopila métricas operativas de varios servicios para generar paneles personalizados que le permiten monitorear el rendimiento y el estado operativo de la solución.

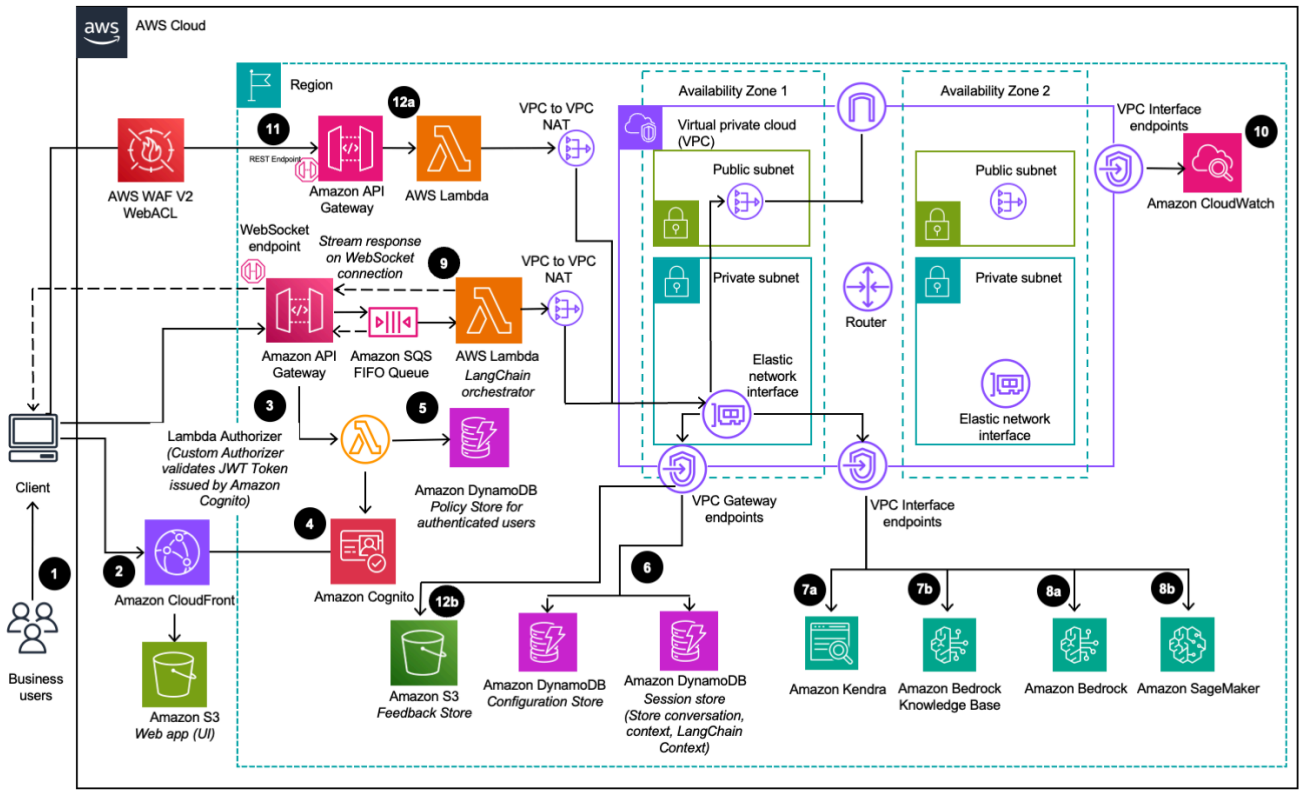
> **Note**
>
> - Si decide implementar esta solución en una Amazon VPC, los datos se enrutarán dentro de su red privada.
> - Si bien el panel de implementación se puede lanzar en la mayoría de las regiones de AWS, los casos de uso implementados tienen ciertas restricciones según la disponibilidad del servicio. Consulte las regiones de AWS compatibles para obtener más información.

Caso de uso de texto

Describe la arquitectura de casos de uso de Text (cuando se implementa con la opción VPC deshabilitada)



Describe la arquitectura de casos de uso de Text (cuando se implementa con la opción VPC habilitada)



El flujo de proceso de alto nivel para los componentes de la solución implementados con la CloudFormation plantilla de AWS es el siguiente:

1. Los usuarios administradores implementan el caso de uso mediante el panel de implementación. Los usuarios empresariales inician sesión en la interfaz de usuario del caso de uso.
2. CloudFront ofrece la interfaz de usuario web que está alojada en un bucket de S3.
3. La interfaz de usuario web aprovecha una WebSocket integración creada mediante API Gateway. La API Gateway está respaldada por una función de autorización Lambda personalizada, que devuelve la política de AWS Identity and Access Management (IAM) correspondiente en función del grupo de Amazon Cognito al que pertenece el usuario autenticador. La política se almacena en DynamoDB.
4. Amazon Cognito autentica a los usuarios y respalda tanto la interfaz de usuario CloudFront web como la API Gateway.
5. Las solicitudes entrantes del usuario empresarial se transfieren de API Gateway a una cola de Amazon SQS y, después, al Orchestrator. LangChain El LangChain Orchestrator es un conjunto de funciones y capas de Lambda que proporcionan la lógica empresarial necesaria para cumplir con las solicitudes del usuario empresarial. La cola permite el funcionamiento asíncrono de la integración entre API Gateway y Lambda. La cola pasa la información de conexión a las funciones

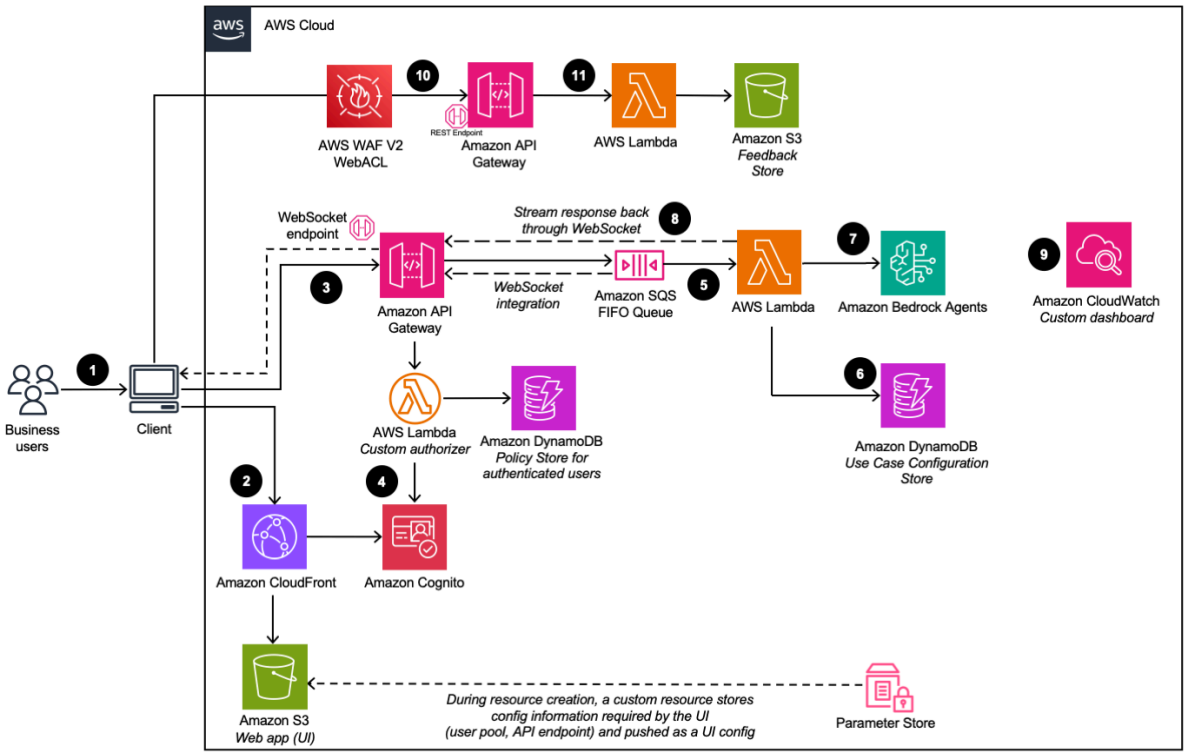
de Lambda, que luego publicarán los resultados directamente en la conexión websocket de API Gateway para admitir llamadas de inferencia de larga duración.

6. LangChain Orchestrator utiliza Amazon DynamoDB para obtener las opciones de LLM configuradas y la información de sesión necesaria (como el historial de chats).
7. Si la implementación tiene una base de conocimientos habilitada, LangChain Orchestrator utiliza Amazon Kendra o Knowledge Bases for Amazon Bedrock para ejecutar una consulta de búsqueda y recuperar extractos de documentos.
8. Con el historial de chat, la consulta y el contexto de la base de conocimientos, el LangChain Orchestrator crea el mensaje final y lo envía al LLM alojado en Amazon Bedrock o Amazon AI. SageMaker
9. Cuando la respuesta proviene del LLM, el LangChain Orchestrator transmite la respuesta a través de la API Gateway WebSocket para que la utilice la aplicación cliente.
10. Con Amazon CloudWatch, esta solución recopila métricas operativas de varios servicios para generar paneles personalizados que le permiten monitorear el rendimiento y el estado operativo de la implementación.
11. Si la recopilación de comentarios está habilitada, se pone a disposición un punto final de la API REST, que aprovecha Amazon API Gateway, para recopilar los comentarios de los usuarios.
12. Los comentarios que respaldan a Lambda aumentan los comentarios enviados con metadatos adicionales específicos para cada caso de uso (por ejemplo, el modelo utilizado) y almacenan los datos en Amazon S3 para que los usuarios los DevOps analicen e informen posteriormente.

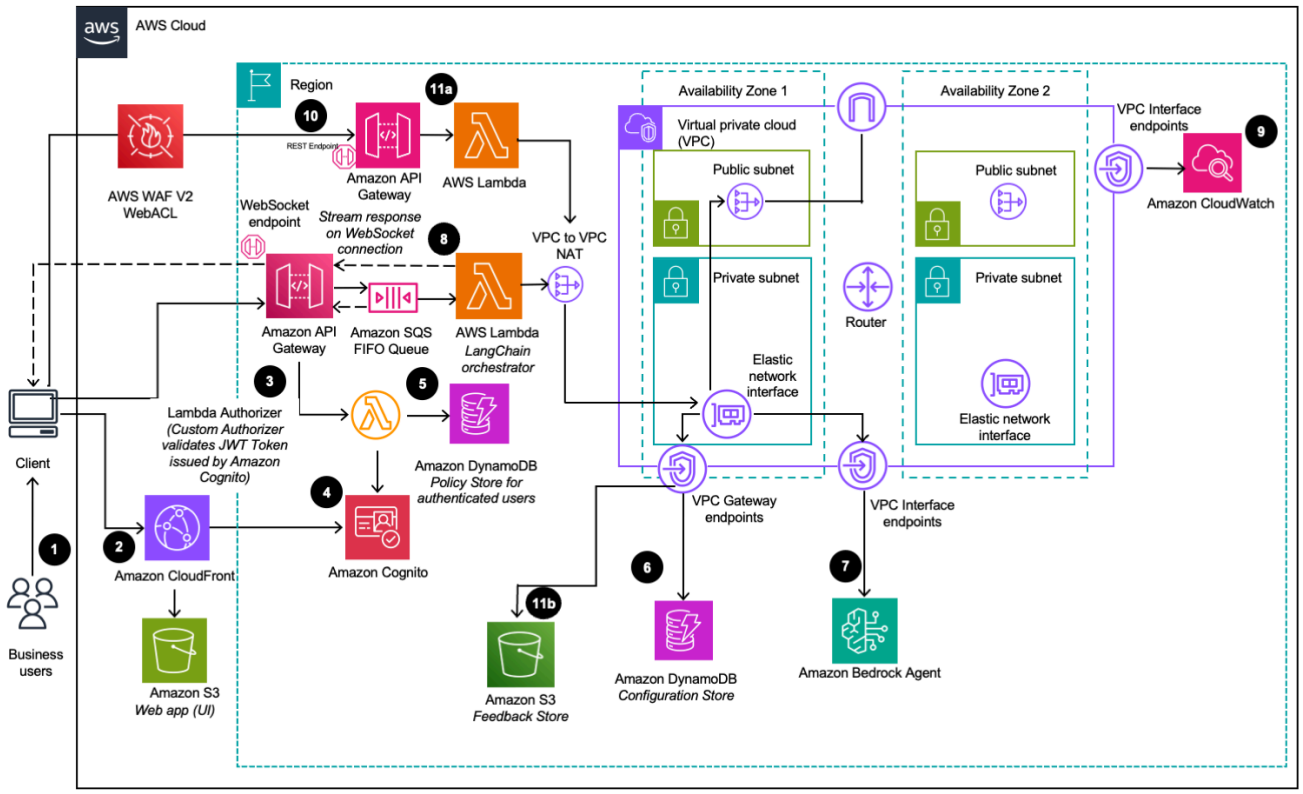
> **Note**
>
> Si decide implementar esta solución en una Amazon VPC, los datos se enrutarán a su red privada.

Caso de uso de Bedrock Agent

Describe la arquitectura de casos de uso de Bedrock Agent (cuando se implementa con la opción VPC deshabilitada)



Describe la arquitectura de casos de uso de Bedrock Agent (cuando se implementa con la opción VPC habilitada)



El flujo de proceso de alto nivel para los componentes de la solución implementados con la CloudFormation plantilla de AWS es el siguiente:

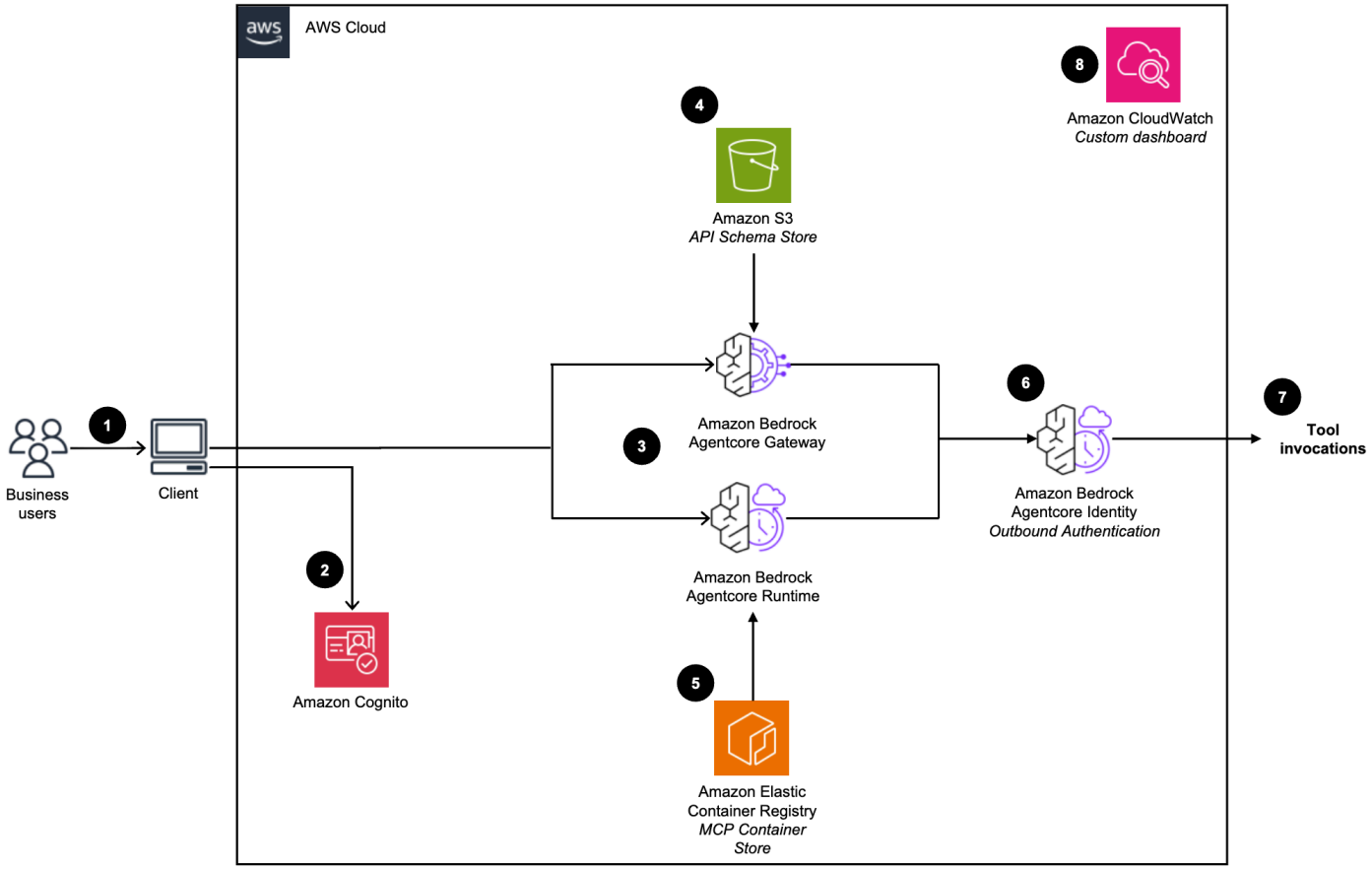
1. Los usuarios administradores implementan el caso de uso mediante el panel de implementación. Los usuarios empresariales inician sesión en la interfaz de usuario del caso de uso.
2. CloudFront proporciona la interfaz de usuario web que está alojada en un bucket de S3.
3. La interfaz de usuario web aprovecha una WebSocket integración creada mediante API Gateway. La API Gateway está respaldada por una función de autorización Lambda personalizada, que devuelve la política de AWS Identity and Access Management (IAM) correspondiente en función del grupo de Amazon Cognito al que pertenece el usuario autenticador. La política se almacena en DynamoDB.
4. Amazon Cognito autentica a los usuarios y respalda tanto la interfaz de usuario CloudFront web como la API Gateway.
5. Las solicitudes entrantes del usuario empresarial se transfieren de API Gateway a una cola de Amazon SQS y, a continuación, a la función AWS Lambda. La cola permite el funcionamiento asíncrono de la integración entre API Gateway y Lambda. La cola pasa la información de conexión a la función Lambda, que luego publicará los resultados directamente en la conexión websocket de API Gateway para admitir llamadas de inferencia de larga duración.

6. La función AWS Lambda usa Amazon DynamoDB para obtener las configuraciones de los casos de uso según sea necesario
7. Con la entrada del usuario y cualquier configuración de caso de uso relevante, la función AWS Lambda crea y envía una carga útil de solicitud al Amazon Bedrock Agent configurado para cumplir con la intención del usuario.
8. Cuando la respuesta proviene del Amazon Bedrock Agent, la función Lambda transmite la respuesta a través de la API WebSocket Gateway para que la utilice la aplicación cliente.
9. Con Amazon CloudWatch, esta solución recopila métricas operativas de varios servicios para generar paneles personalizados que le permiten monitorear el rendimiento y el estado operativo de la implementación.
10. Si la recopilación de comentarios está habilitada, se pone a disposición un punto final de la API REST, que aprovecha Amazon API Gateway, para recopilar los comentarios de los usuarios.
11. Los comentarios que respaldan a Lambda aumentan los comentarios enviados con metadatos adicionales específicos para cada caso de uso y almacenan los datos en Amazon S3 para que los usuarios los DevOps analicen e informen posteriormente.

> **Note**
>
> Si decide implementar esta solución en una Amazon VPC, los datos se enrutarán dentro de su red privada.

Caso de uso del servidor MCP

Describe la arquitectura de casos de uso del servidor MCP



El caso de uso del servidor MCP permite la implementación y la administración de servidores del Model Context Protocol en Amazon Bedrock AgentCore. Los servidores MCP proporcionan una interfaz estandarizada para que las aplicaciones de IA accedan a las herramientas, los recursos y las fuentes de datos empresariales.

La solución admite dos métodos de implementación:

- Método de puerta de enlace: agrupa las funciones Lambda, APIs REST o servidores MCP externos existentes como herramientas MCP, gestionando la traducción de protocolos automáticamente
- Método de ejecución: despliega servidores MCP en contenedores personalizados a partir de imágenes de Amazon ECR

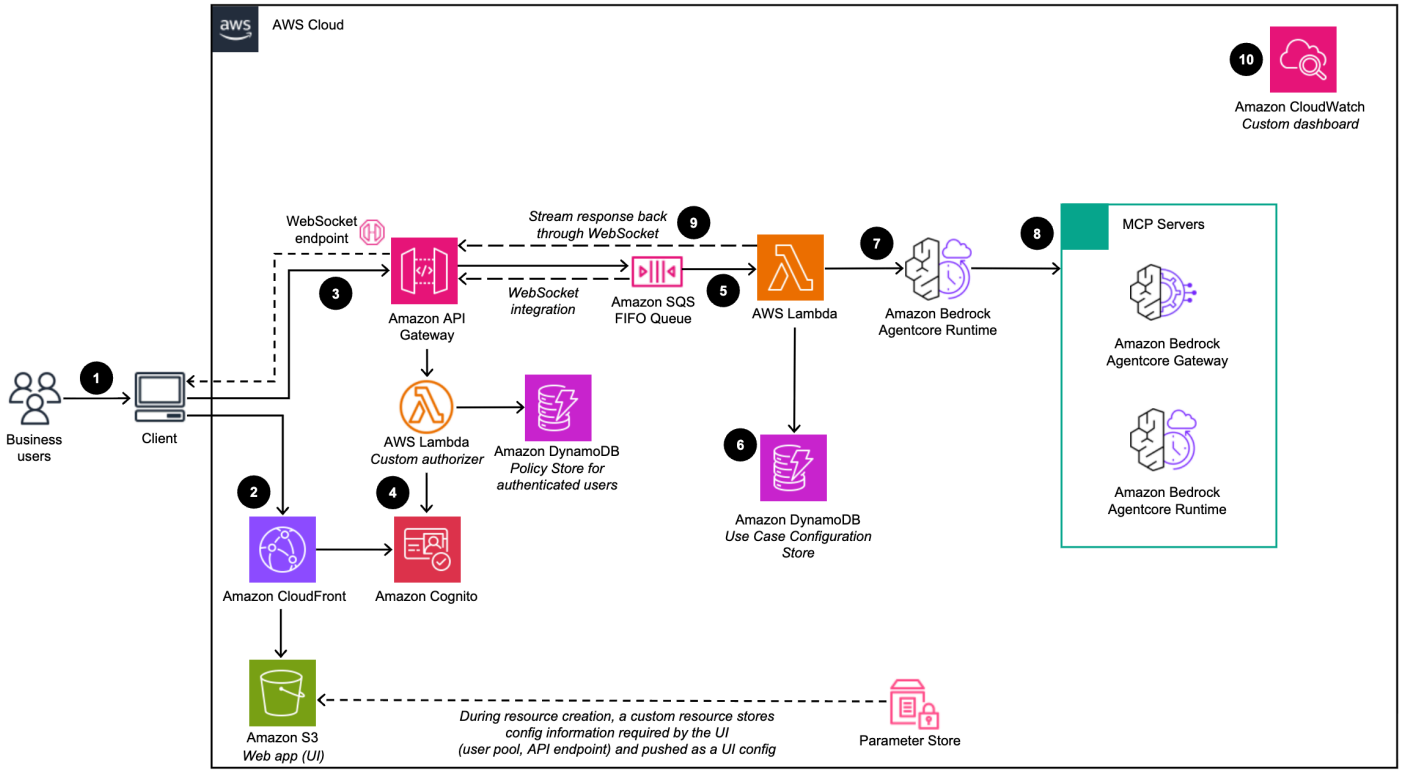
El flujo de proceso de alto nivel para la implementación del servidor MCP es el siguiente:

1. Los usuarios administradores implementan el caso de uso del servidor MCP mediante el panel de implementación y seleccionan el método de implementación Gateway o Runtime.

2. Esta acción se autentica con Amazon Cognito.
3. Para la implementación de Gateway, la solución crea un Amazon Bedrock AgentCore Gateway que transforma las funciones APIs Lambda existentes o los servidores MCP externos en herramientas compatibles con MCP. Para la implementación en tiempo de ejecución, la solución implementa servidores MCP en contenedores en Amazon Bedrock AgentCore Runtime mediante las imágenes ECR proporcionadas.
4. Las implementaciones de Gateway recuperan los API/Lambda/Smithy esquemas necesarios de la ubicación en la que se cargaron en Amazon S3 o se conectan directamente a los puntos de enlace URL del servidor MCP.
5. Las implementaciones en tiempo de ejecución recuperan el servidor MCP en contenedores proporcionado por el usuario de Amazon Elastic Container Registry (ECR)
6. El servidor MCP está equipado con un cliente Amazon Bedrock Identity AgentCore OAuth
7. El servidor MCP hace que las herramientas asociadas estén disponibles en el punto final /mcp para que las descubran los agentes.
8. Amazon CloudWatch recopila métricas y registros operativos de las implementaciones de servidores MCP para su supervisión y solución de problemas.

Caso de uso de Agent Builder

Describe la arquitectura de Agent Builder



El flujo de proceso de alto nivel para los componentes de Agent Builder implementados con la CloudFormation plantilla de AWS es el siguiente:

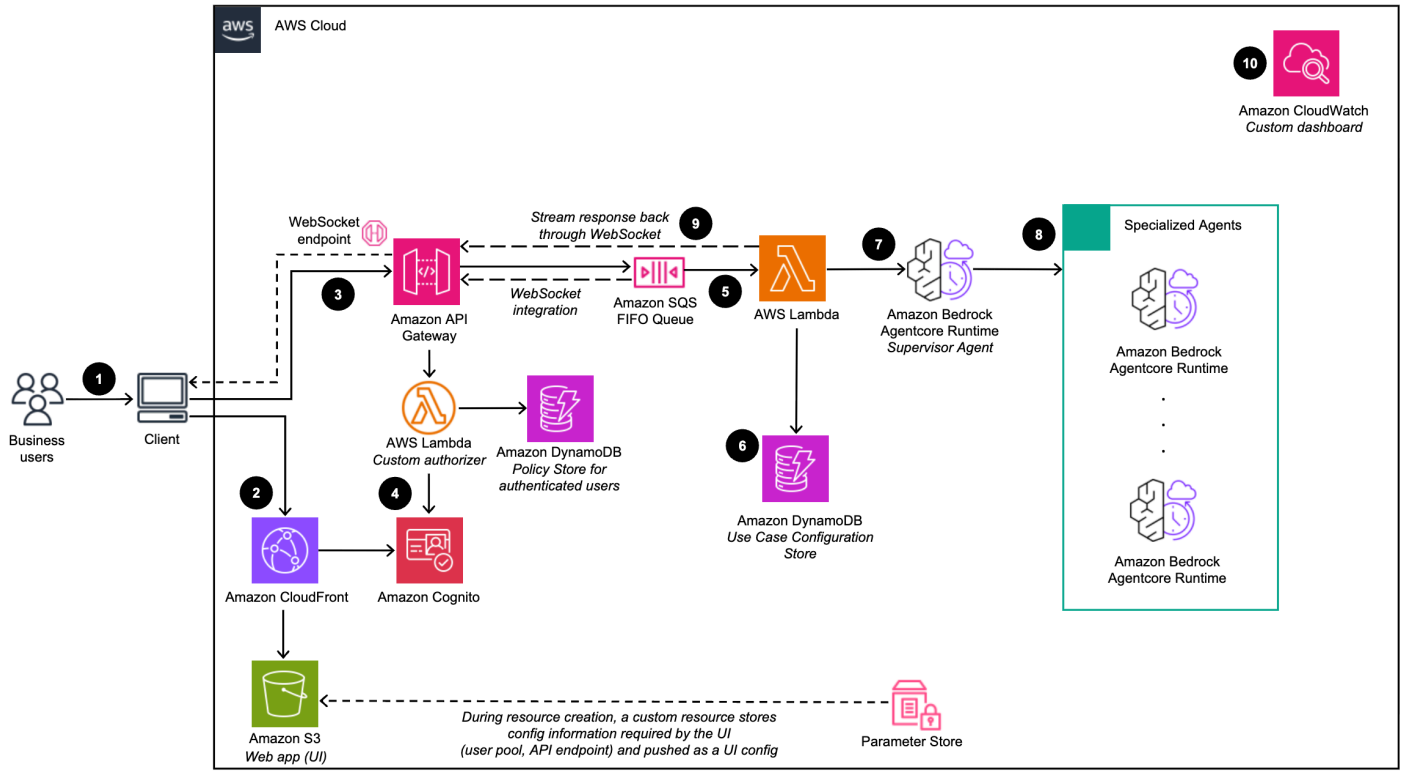
1. Los usuarios administradores implementan el caso de uso mediante el panel de implementación. Los usuarios empresariales inician sesión en la interfaz de usuario del caso de uso.
2. CloudFront proporciona la interfaz de usuario web que está alojada en un bucket de S3.
3. La interfaz de usuario web aprovecha una WebSocket integración creada mediante API Gateway. La API Gateway está respaldada por una función de autorización Lambda personalizada, que devuelve la política de AWS Identity and Access Management (IAM) correspondiente en función del grupo de Amazon Cognito al que pertenece el usuario autenticador. La política se almacena en DynamoDB.
4. Amazon Cognito autentica a los usuarios y respalda tanto la interfaz de usuario CloudFront web como la API Gateway.
5. Las solicitudes entrantes del usuario empresarial se transfieren de API Gateway a una cola de Amazon SQS y, a continuación, a la función AWS Lambda. La cola permite el funcionamiento asíncrono de la integración entre API Gateway y Lambda. La cola pasa la información de conexión a la función Lambda, que luego publicará los resultados directamente en la conexión websocket de API Gateway para admitir llamadas de inferencia de larga duración.

6. La función AWS Lambda recupera la configuración del agente de DynamoDB.
7. Con la entrada del usuario y cualquier configuración de caso de uso relevante, la función AWS Lambda crea y envía una carga útil de solicitud al agente, que se ejecuta en Amazon Bedrock Runtime. AgentCore
8. El agente se conecta a los servidores MCP asociados y registra las herramientas en la instancia del agente de Strands. A continuación, el agente selecciona y ejecuta las acciones de forma autónoma en función de las descripciones de las herramientas y los requisitos de las tareas.
9. Cuando la respuesta regresa del entorno de AgentCore ejecución de Amazon Bedrock, la función Lambda transmite la respuesta a través de la API WebSocket Gateway para que la utilice la aplicación cliente.

> **Note**
>
> - El procesamiento del agente está limitado al tiempo de espera de ejecución de Lambda (15 minutos).

Caso de uso de Workflow Builder

Describe la arquitectura de Workflow Builder



El flujo de proceso de alto nivel para los componentes de Workflow Builder implementados con la CloudFormation plantilla de AWS es el siguiente:

1. Los usuarios administradores implementan el flujo de trabajo mediante el panel de implementación y seleccionan los agentes de Agent Builder para incluirlos como agentes especializados.
2. CloudFront ofrece la interfaz de usuario web que está alojada en un bucket de S3.
3. La interfaz de usuario web aprovecha una WebSocket integración creada mediante API Gateway. La API Gateway está respaldada por una función de autorización Lambda personalizada, que devuelve la política de AWS Identity and Access Management (IAM) correspondiente en función del grupo de Amazon Cognito al que pertenece el usuario autenticador. La política se almacena en DynamoDB.
4. Amazon Cognito autentica a los usuarios y respalda tanto la interfaz de usuario CloudFront web como la API Gateway.
5. Las solicitudes entrantes del usuario empresarial se transfieren de API Gateway a una cola de Amazon SQS y, a continuación, a la función AWS Lambda. La cola permite el funcionamiento asíncrono de la integración entre API Gateway y Lambda.
6. La función AWS Lambda recupera la configuración del flujo de trabajo de DynamoDB, incluida la lista de agentes especializados en Agent Builder.

7. Con las entradas del usuario y la configuración del flujo de trabajo, Lambda envía las solicitudes al Amazon Bedrock AgentCore Runtime que aloja al agente supervisor.
8. El agente supervisor crea instancias locales de todos los agentes especializados de Agent Builder dentro del AgentCore entorno de ejecución. Estos agentes especializados se registran como herramientas mediante el patrón Agentes como herramientas. Luego, el supervisor selecciona y delega el trabajo de forma autónoma en agentes especializados en función de las descripciones de los agentes y los requisitos de la tarea.
9. El agente supervisor agrega los resultados de los agentes especializados y formula la respuesta final y la devuelve a la Lambda para que se transmita a la aplicación cliente a través del API Gateway Websocket.

> **Note**
>
> - El procesamiento del flujo de trabajo está limitado al tiempo de espera de ejecución de Lambda (15 minutos).

Consideraciones sobre el diseño de AWS Well-Architected

Esta solución se diseñó con las mejores prácticas del AWS Well-Architected Framework, que ayuda a los clientes a diseñar y operar cargas de trabajo confiables, seguras, eficientes y rentables en la nube.

En esta sección se describe la aplicación de los principios de diseño y las prácticas recomendadas del Marco de Well-Architected al crear esta solución.

Excelencia operativa

En esta sección se describe cómo diseñamos esta solución utilizando los principios y las prácticas recomendadas del pilar de excelencia operativa.

- Creamos la solución infrastructure-as-code con Amazon CloudFormation.
- Las funciones de Lambda envían métricas personalizadas CloudWatch y un CloudWatch panel personalizado para supervisar el estado de la solución.
- Los componentes de la solución están altamente modularizados, lo que proporciona la flexibilidad necesaria para elegir los componentes que se van a implementar.

Seguridad

En esta sección se describe cómo diseñamos esta solución utilizando los principios y las prácticas recomendadas del pilar de seguridad.

- El panel de implementación y todos los casos de uso están autenticados y autorizados con Amazon Cognito.
- Todas las comunicaciones entre servicios utilizan las funciones de IAM de AWS.
- Todas las funciones de la solución se basan en el acceso con el mínimo privilegio, es decir, solo se conceden los permisos mínimos necesarios.
- Todo el almacenamiento de datos, incluidos los buckets S3, DynamoDB y Amazon Kendra, tiene cifrado en reposo.

Fiabilidad

En esta sección se describe cómo diseñamos esta solución utilizando los principios y las prácticas recomendadas del pilar de fiabilidad.

- Arquitectura basada en el paradigma sin servidor.
- Creamos la arquitectura para ofrecer escalabilidad horizontal bajo demanda y recuperación automática en caso de fallo de la infraestructura subyacente.
- La arquitectura incluye el almacenamiento en búfer y la limitación de las solicitudes para no sobrecargar los puntos finales subyacentes.

Eficiencia del rendimiento

En esta sección se describe cómo diseñamos esta solución utilizando los principios y las prácticas recomendadas del pilar de eficiencia del rendimiento.

- La solución utiliza DynamoDB, una base de datos NoSQL sin servidor totalmente gestionada con escalado bajo demanda.
- La solución utiliza Amazon S3 para el almacenamiento de objetos y para alojar un sitio web (mediante CloudFront) a fin de ofrecer un bajo coste, escalable y una durabilidad de 11 9 segundos.

Optimización de costos

En esta sección se describe cómo diseñamos esta solución utilizando los principios y las prácticas recomendadas del pilar de optimización de costos.

- Siempre que fue posible, diseñamos la solución para usar una arquitectura sin servidor, de modo que solo paga por lo que usa.

Sostenibilidad

En esta sección se describe cómo diseñamos esta solución utilizando los principios y las mejores prácticas del pilar de sostenibilidad.

- La arquitectura modular y dividida en componentes de la solución ofrece la flexibilidad de personalizar los recursos que se van a aprovisionar para casos de uso individuales.
- La arquitectura utiliza computación y almacenamiento sin servidor, lo que optimiza la utilización de los recursos.
- Como solución basada en la nube, esta solución se beneficia de los recursos compartidos, las redes, la alimentación, la refrigeración y las instalaciones físicas.

Detalles de la arquitectura

En esta sección se describen los componentes y los servicios de AWS que componen esta solución y los detalles de la arquitectura sobre cómo funcionan juntos estos componentes.

Los servicios de AWS en esta solución

Servicio de AWS	Description (Descripción)
Amazon API Gateway	Principal. Este servicio proporciona el REST APIs para el panel de implementación y la WebSocket API para el caso de uso.
AWS CloudFormation	Principal. Esta solución se distribuye como una CloudFormation plantilla e CloudForm ation implementa los recursos de AWS para la solución.
Amazon CloudFront	Núcleo. CloudFront sirve el contenido web alojado en Amazon S3.
Amazon Cognito	Principal. Este servicio se encarga de la administración de usuarios y la autenticación de la API.
Amazon DynamoDB	Principal. DynamoDB almacena la informaci ón de implementación y los detalles de configuración para el panel de implementación. Almacena el historial de chat y las conversac iones IDs en el caso de uso de Text para permitir la desambiguación del historial de conversaciones y las consultas.
AWS Lambda	Principal. La solución utiliza funciones Lambda para: * Respalde los puntos finales de WebSocket REST y API * Gestione la lógica básica

Servicio de AWS	Description (Descripción)
	de cada orquestador de casos de uso * Implemente recursos personalizados durante la implementación CloudFormation
Amazon S3	Principal. Amazon S3 aloja el contenido web estático.
Amazon CloudWatch	Admite. Esta solución publica los registros de los recursos de la solución en CloudWatch los registros y publica las métricas en las CloudWatch métricas. La solución también crea un CloudWatch panel para ver estos datos.
AWS Systems Manager	Admite. Systems Manager proporciona monitoreo de recursos a nivel de aplicación y visualización de operaciones de recursos y datos de costos. También se utiliza para almacenar datos de configuración en el almacén de parámetros.
AWS WAF	Admite. AWS WAF se implementa delante de la implementación de API Gateway para protegerlo.
Amazon Bedrock	Opcional. La solución aprovecha Amazon Bedrock para acceder a modelos básicos o personalizados, a Amazon Bedrock Agents y a las bases de conocimiento de Amazon Bedrock. Amazon Bedrock es la integración recomendada para evitar que sus datos salgan de la red de AWS.
Amazon Bedrock AgentCore	Opcional: la solución aprovecha Amazon Bedrock AgentCore para ejecutar y dar soporte a las conexiones del servidor MCP, así como a los casos de uso de Agent Builder y Workflow.

Servicio de AWS	Description (Descripción)
Amazon Elastic Container Registry (Amazon ECR)	Opcional. Para las implementaciones de Agent Builder, ECR almacena y distribuye imágenes de contenedores de agentes. La solución utiliza la memoria caché Pull-Through de ECR para recuperar automáticamente las imágenes de los agentes prediseñadas del repositorio de ECR público del equipo de la GAAB.
AWS Distro para OpenTelemetry (ADOT)	Opcional. Para las implementaciones de Agent Builder, ADOT proporciona una instrumen tación automática para la observabilidad de los agentes, lo que permite el rastreo distribuido y el registro estructurado de las operaciones de los agentes.
Amazon Kendra	Opcional. En el caso de uso de Text, los usuarios administradores pueden optar por conectar un índice de Amazon Kendra para usarlo como base de conocimientos para la conversación con el LLM. Esto se puede utilizar para introducir nueva información en el LLM, lo que le permite utilizar esa información en sus respuestas.

Servicio de AWS	Description (Descripción)
Amazon SageMaker AI	Opcional. La solución se puede integrar con un punto final de inferencia de Amazon SageMaker AI al FMs que acceder y que esté alojado en su cuenta y región de AWS, y es una integración preferida para evitar que sus datos salgan de la red de AWS. **Note** Debe implementar la solución en la misma región en la que está disponible el punto final de inferencia.
Amazon Virtual Private Cloud	Opcional. La solución ofrece la opción de implementar componentes con una configura ción habilitada para VPC. Al implementar la solución con una configuración habilitad a para VPC, tiene la opción de permitir que la solución cree una VPC para usted o usar una VPC existente que exista en la misma cuenta y región en la que se implementará la solución (traiga su propia VPC). Si la solución crea la VPC, crea los componentes de red necesarios, que incluyen subredes, grupos de seguridad y sus reglas, tablas de enrutamiento, red, puertas de enlace NAT ACLs, puertas de enlace de Internet, puntos de enlace de VPC y sus políticas.

Panel de implementación

Autorizadores personalizados de API Gateway

A simple vista, los autorizadores personalizados de Lambda para API Gateway se utilizan para todas las llamadas a la API (RESTful tanto las llamadas como las WebSocket basadas) a fin de validar si un usuario determinado tiene permiso para realizar una acción en función de los grupos a los que pertenece. Este autorizador personalizado está respaldado por una tabla de DynamoDB que contiene las políticas de cada grupo. Al invocar una API, API Gateway invoca la función Lambda de autorización personalizada, que decodifica el token de acceso de Amazon Cognito proporcionado para determinar a qué grupos de usuarios pertenece el usuario. A continuación, se consulta la tabla de políticas por nombre de grupo para obtener la política correspondiente a ese grupo.

En cada implementación de un nuevo caso de uso, la política de administración se actualiza para almacenar una nueva declaración que permita la acción Execute-API:Invoke en la API de ese caso de uso. Cuando se eliminan los casos de uso, la declaración correspondiente se elimina de la política.

En el caso de los grupos creados para un caso de uso individual, solo hay una sentencia en la política, lo que permite la acción Execute-API:Invoke únicamente en la API de ese caso de uso.

Gracias a esta estructura, cualquier usuario que pertenezca al grupo de un caso de uso puede acceder a la API de ese caso de uso. Un solo usuario también se puede añadir manualmente a varios grupos para permitir que ese usuario utilice varios casos de uso.

> **Warning**
>
> También puede editar manualmente las políticas de un grupo determinado en la tabla de políticas si desea conceder acceso a un nuevo caso de uso a un grupo de usuarios existente. El grupo de casos de uso se elimina cuando se elimina el caso de uso (incluso si ha realizado modificaciones manuales), así que proceda con precaución al eliminar un caso de uso.

En el caso de que una pila de casos de uso se implemente de forma independiente (sin el uso del panel de implementación), se crea un grupo de usuarios de Amazon Cognito para esa implementación que contiene un solo usuario con acceso a la API de ese caso de uso. Este grupo de usuarios pertenece únicamente a este caso de uso y no se comparte con otras implementaciones independientes.

Caso de uso de texto

Soporte de streaming

En una aplicación de chat, la latencia es una métrica importante para permitir una experiencia de usuario responsiva. La posibilidad de que las inferencias de LLM tarden de segundos a minutos plantea dificultades a la hora de ofrecer mejor el contenido a los clientes. Por este motivo, varios proveedores de LLM permiten transmitir las respuestas a la persona que llama. En lugar de esperar a que se complete toda la inferencia antes de devolver una respuesta, se puede devolver cada token cuando esté disponible.

Para respaldar el uso de esta función, el caso de uso de Text se ha diseñado para utilizar una WebSocket API que respalde la experiencia de chat. Esto WebSocket se implementa a través de API Gateway. El uso de una WebSocket API permite crear una conexión al principio de una sesión de chat y transmitir las respuestas a través de ese conector. Esto permite que las aplicaciones frontend proporcionen una mejor experiencia de usuario.

> **Note**
>
> Incluso si un modelo ofrece soporte de streaming, esto no significa necesariamente que la solución pueda transmitir las respuestas a través de la WebSocket API. Es necesario que la solución habilite una lógica personalizada para admitir la transmisión para cada proveedor de modelos. Si la transmisión está disponible, los usuarios administradores podrán utilizar enable/disable esta función en el momento de la implementación.

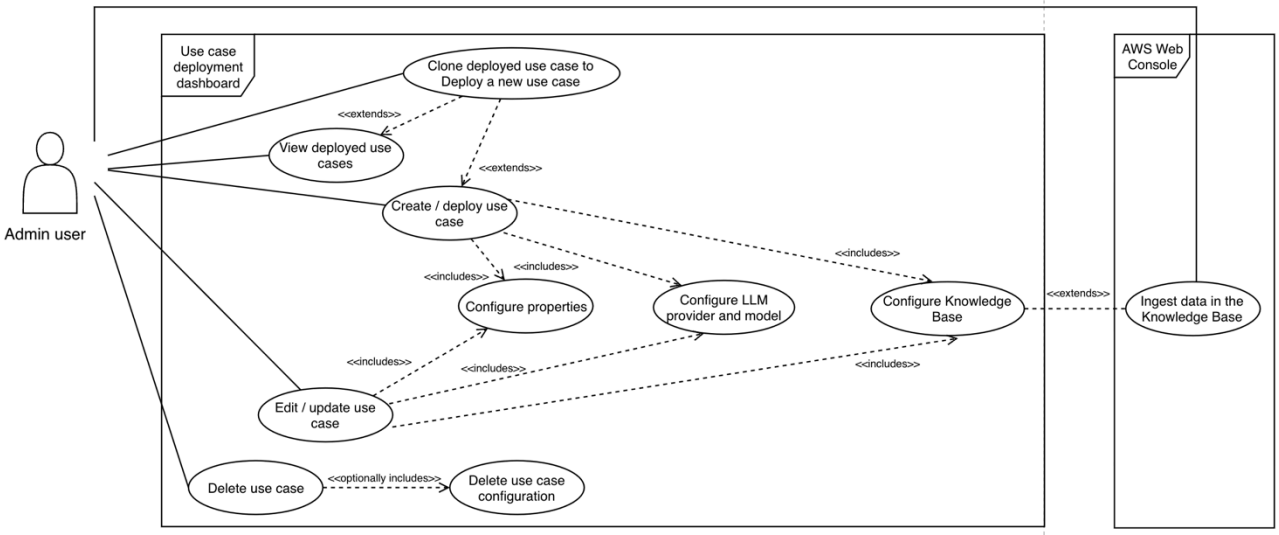
Cómo funciona la solución Generative AI Application Builder en AWS

El usuario administrador interactúa principalmente con el panel de implementación para ver, crear y administrar las implementaciones de casos de uso nuevas y existentes. A través de este panel, el usuario administrador tiene acceso a las siguientes acciones:

- Ver la lista de despliegues
- Cree nuevos despliegues
- Edite las implementaciones existentes

- Clona la configuración de una implementación para crear una nueva implementación
- Eliminar una implementación (desaprovisionar los recursos mediante una CloudFormation eliminación)
- Elimine permanentemente los detalles de configuración de una implementación

Muestra un diagrama de casos de uso para el usuario administrador del panel de implementación

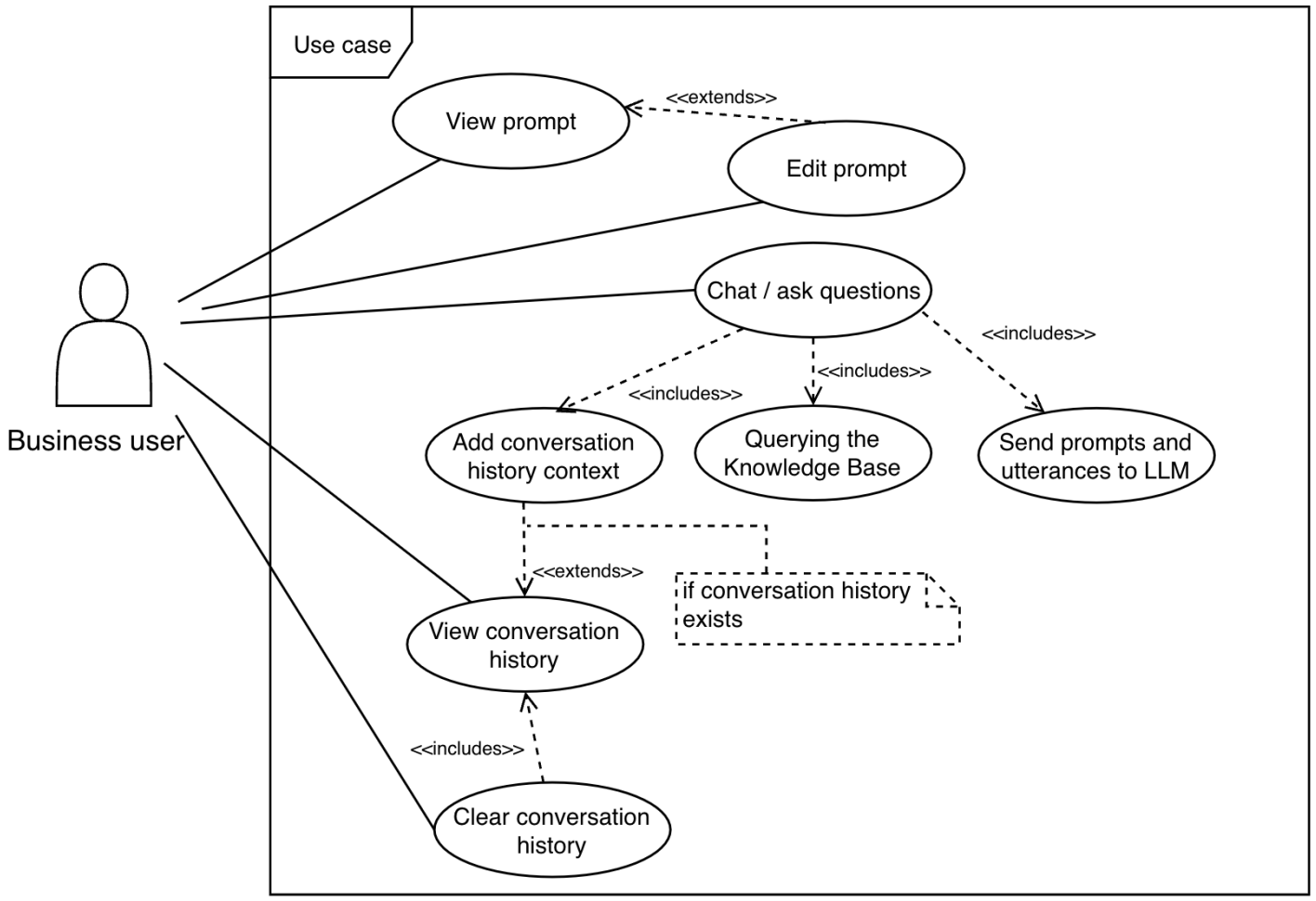


> **Note**
>
> Es posible que el usuario administrador no tenga acceso directo a la consola de AWS. En ese caso, el usuario administrador debe trabajar con el DevOps usuario para respaldar acciones como la ingesta de datos en una base de conocimiento de Kendra.

En el caso de uso de Text, el usuario empresarial tiene acceso a una interfaz de usuario que le permite chatear con el LLM. Los detalles de esta configuración se controlan mediante los ajustes de implementación configurados por el usuario administrador. En el caso de uso de Text, el usuario empresarial tiene acceso a las siguientes acciones:

- Envíe mensajes a través de la interfaz de chat
- Ver el historial de conversaciones
- Borra el historial de conversaciones
- Ver mensaje
- Solicitud de edición

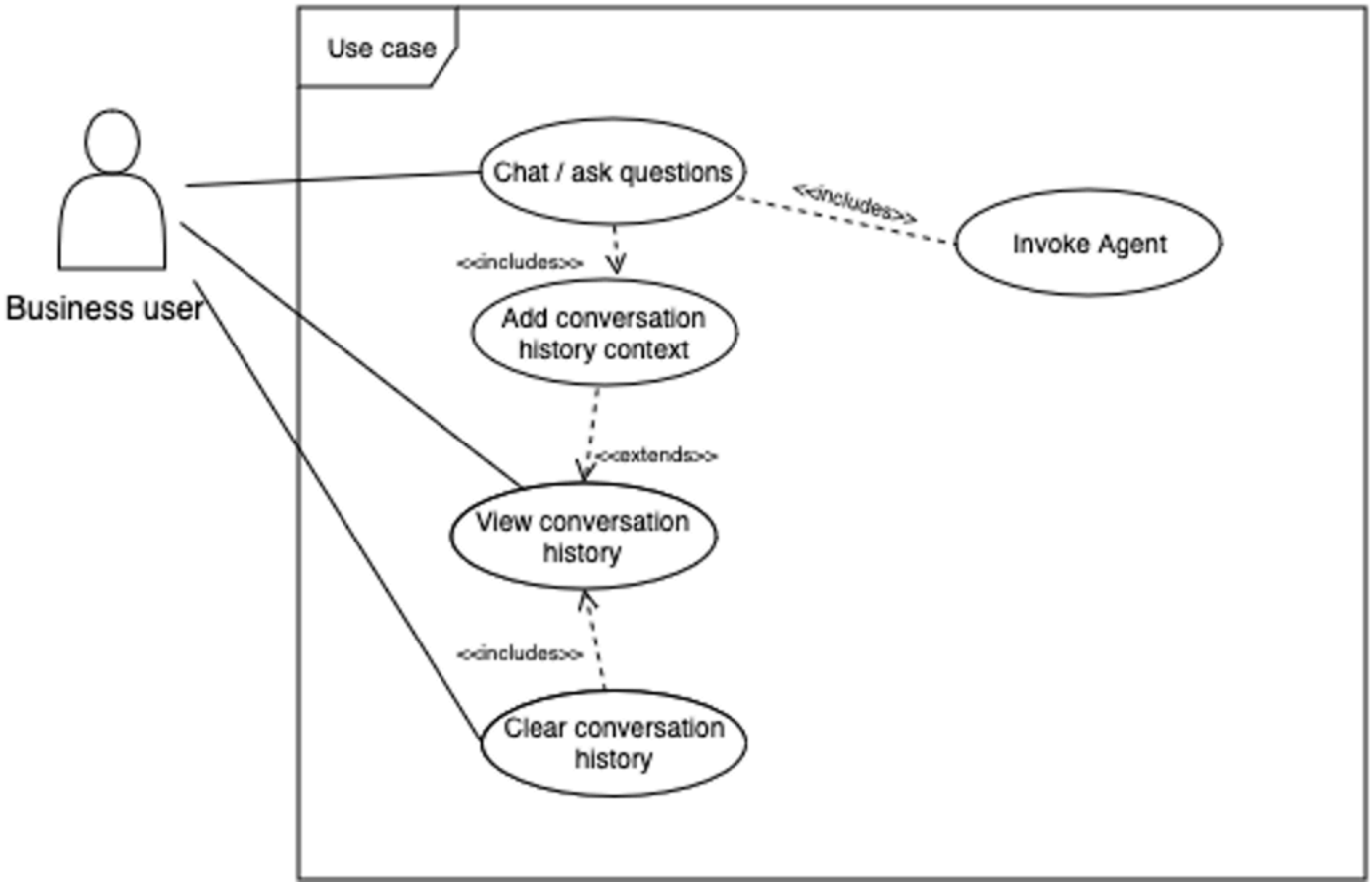
Muestra un diagrama de casos de uso para el usuario empresarial del caso de uso de Text



Con el caso de uso de Bedrock Agent, el usuario empresarial puede acceder a una interfaz de usuario para chatear con el agente de Amazon Bedrock configurado. El usuario administrador puede configurar estos detalles en los ajustes de implementación. En el caso de uso de Bedrock Agent, el usuario empresarial tiene acceso a las siguientes acciones:

- Envíe mensajes a través de la interfaz de chat
- Ver el historial de conversaciones
- Borra el historial de conversaciones

Muestra un diagrama de casos de uso para el usuario empresarial del caso de uso de Bedrock Agent



Agent Builder

Agent Builder proporciona una plataforma para crear, implementar y administrar agentes de IA listos para la producción en Amazon Bedrock. AgentCore En esta sección se describen los componentes técnicos y los detalles de la implementación.

AgentCore integración

Agent Builder utiliza un enfoque de despliegue basado en la configuración con imágenes de agentes prediseñadas para permitir despliegues de agentes rápidos, seguros y escalables.

Imágenes de agentes prediseñadas

El equipo de la GAAB crea las imágenes de los contenedores de agentes durante el CI/CD proceso y las publica en un repositorio de ECR público. Cada versión de imagen está vinculada a la versión

de la solución GAAB (por ejemplo, v4.0.0 →:v4.0.0). gaab-strands-agent Las imágenes se basan en el SDK de Strands e incluyen:

- Entorno de ejecución del agente
- Integración de clientes MCP
- Capacidades de administración de memoria
- OpenTelemetry instrumentación

Caché desplegable ECR

La solución utiliza la memoria caché Pull-Through de ECR para distribuir automáticamente las imágenes de los agentes desde el repositorio de ECR público al ECR privado del cliente. Este servicio gestionado por AWS:

- Almacena en caché las imágenes al extraerlas por primera vez (retraso de 2 a 5 minutos)
- Elimina la lógica de copia de imágenes personalizada
- Proporciona disponibilidad de imágenes locales para despliegues posteriores
- Crea reglas de caché únicas por implementación para evitar conflictos

Almacenamiento de configuraciones

Las configuraciones de los agentes se almacenan en DynamoDB junto con las configuraciones de casos de uso existentes. Cada configuración incluye:

- Plantilla de solicitud del sistema
- Proveedor de modelos e ID de modelo
- Parámetros del modelo (temperatura, max_tokens)
- Referencias y puntos finales del servidor MCP
- Configuración de memoria (conmutador de memoria a largo plazo)
- Metadatos de despliegue

Registro de versiones de imágenes

Una tabla de DynamoDB rastrea las versiones de las imágenes de los agentes disponibles y su URIs caché, lo que permite la administración de versiones y la compatibilidad con versiones anteriores.

Configuración del agente

Indicaciones del sistema

Las indicaciones del sistema definen el comportamiento, la personalidad y las capacidades del agente. Los usuarios administradores pueden:

- Edite la plantilla predeterminada a través de la interfaz de usuario de Agent Builder
- Incluya instrucciones para el uso de la herramienta y el formato de las respuestas
- Restablezca la plantilla predeterminada en cualquier momento

Selección de modelos

Agent Builder es compatible con los modelos de Amazon Bedrock en la versión 4.0.0:

- Proveedor del modelo: Amazon Bedrock (solo opción en la versión 4.0.0)
- Selección de modelos: Claude, Nova y otros modelos de Bedrock
- Parámetros del modelo: temperatura, max_tokens, top_p y ajustes específicos del modelo

Integración del servidor MCP

Los servidores Model Context Protocol proporcionan a los agentes acceso a herramientas y datos empresariales:

- Detección de servidores mediante el punto final de la API GET/mcp
- Configuración dinámica sin cambios de código
- Autenticación y administración de terminales
- Capacidad de la herramienta: exposición a los agentes

Transmisión y procesamiento

Transmisión en tiempo real

Agent Builder utiliza los eventos enviados por el servidor (SSE) de forma AgentCore puente a transmisión de respuestas en WebSocket tiempo real:

- La función Lambda establece la conexión SSE con Runtime AgentCore

- Las transmisiones se conectan a API Gateway WebSocket
- Permite la entrega de token-by-token respuestas a los clientes
- Mantiene la conexión para las solicitudes de larga duración

Restricciones de procesamiento

El procesamiento del agente en la versión 4.0.0 está limitado al tiempo de espera de ejecución de Lambda:

- Tiempo máximo de procesamiento: 15 minutos
- Modelo de procesamiento sincrónico
- Adecuado para agentes conversacionales y flujos de trabajo moderados
- El soporte asíncrono ampliado está previsto para la versión 4.1 o posterior

Administración de la memoria

Memoria a corto plazo

Habilitada de forma predeterminada para todos los agentes que utilizan una configuración personalizada MemoryHookProvider:

- Captura los eventos de conversación a través de los controladores de devolución de llamadas de Strands
- Se organiza por ActorID y SessionID para aislar el contexto
- Mantiene el contexto de la conversación dentro de las sesiones
- Integración automática con AgentCore Memory

Memoria a largo plazo

Función opcional que utiliza AgentCore la herramienta de memoria de strands_tools:

- Conmutador sencillo en la interfaz de usuario de Agent Builder
- Estrategia de memoria semántica con ajustes predeterminados
- Acceso controlado por agentes mediante la invocación natural de herramientas
- Almacena la información extraída en todas las sesiones

- Usa ConversationID como SessionID

Observabilidad

OpenTelemetry Distribución AWS (ADOT)

Los agentes se instrumentan automáticamente durante la creación del contenedor:

- Generación automática de trazas para las operaciones de los agentes
- Rastreo distribuido entre los límites del servicio
- Registro estructurado con correlación IDs
- Integración con CloudWatch Transaction Search

Flujo de autenticación

Los usuarios se autentican a través de Amazon Cognito con tokens JWT validados por autorizadores Lambda personalizados que recuperan las políticas de IAM de DynamoDB en función de los grupos de usuarios.

Creador de flujos de trabajo

Workflow Builder permite la orquestación de varios agentes mediante la creación de un agente supervisor que coordina varios agentes de Agent Builder mediante el patrón de delegación de agentes como herramientas.

Arquitectura de flujo de trabajo

Componentes clave

- Agente supervisor: agente de punto de entrada que recibe las solicitudes de los usuarios y las delega en agentes especializados
- Agentes especializados: casos de uso de Agent Builder registrados como herramientas para el supervisor
- Registro de agentes: tabla de DynamoDB que almacena las configuraciones y los metadatos de los agentes
- Capa de orquestación: implementa el patrón Agents as Tools en el SDK

Instanciación de agentes

Creación de agentes locales

Todos los agentes especializados se instancian localmente en el mismo tiempo AgentCore de ejecución:

1. Recupera las configuraciones de los agentes de DynamoDB
2. Crea instancias locales de cada agente de Agent Builder
3. Cada agente mantiene sus propias conexiones de servidor MCP
4. El agente supervisor registra a los agentes especializados como herramientas
5. El SDK de Strands gestiona la selección y delegación de agentes

Planificación de la implementación

En esta sección se describen las consideraciones de costo, seguridad, región y cuota a la hora de planificar la implementación.

> **Important**
>
> Esta solución aprovecha Amazon Bedrock como el servicio principal para acceder a los modelos generados por IA. Primero debe solicitar el acceso a los modelos antes de que estén disponibles para su uso en la solución. Para obtener más información, consulte Model access en la Guía del usuario de Amazon Bedrock.

Regiones de AWS admitidas

> **Important**
>
> Esta solución utiliza opcionalmente los servicios Amazon Bedrock y Amazon Kendra, que actualmente no están disponibles en todas las regiones de AWS. Debe lanzar esta solución en una región de AWS en la que estén disponibles estos servicios. Para obtener la disponibilidad más reciente de los servicios de AWS por región, consulte la lista de servicios regionales de AWS.

El generador de aplicaciones de IA generativa en AWS es compatible con las siguientes regiones de AWS:

Nombre de la región	
Este de EE. UU. (Ohio)	Canadá (centro)
Este de EE. UU. (Norte de Virginia)	Europa (Fráncfort)
EE.UU. Oeste (Norte de California)	Europa (Irlanda)
Oeste de EE. UU. (Oregón)	Europa (Londres)
Asia-Pacífico (Mumbai)	Europa (Milán)

Nombre de la región	
Asia-Pacífico (Seúl)	Europa (París)
Asia-Pacífico (Singapur)	Europa (Estocolmo)
Asia-Pacífico (Sídney)	Middle East (Bahrain)
Asia-Pacífico (Tokio)	América del Sur (São Paulo)

> **Note**
>
> Si en sus implementaciones utiliza un modelo básico al que se accede desde fuera de AWS, consulte con el proveedor del modelo en qué regiones APIs están disponibles. Si solo APIs están disponibles en determinadas regiones, es posible que experimente inestabilidad en forma de alta latencia o incluso de tiempos de espera. También es importante que consultes con los equipos legales y de cumplimiento de tu organización para evaluar las posibles consecuencias de que los datos sobrepasen las fronteras regionales.

Costo

Con esta solución de AWS, solo paga por los recursos que utilice y no hay tarifas mínimas ni cargos de configuración. Los usuarios pagan por el panel de control utilizado para lanzar los casos de uso de IA generativa y por cualquier caso de uso que se implemente. El coste de los casos de uso implementados depende de las configuraciones. Ejemplos de configuraciones:

1. Un sencillo panel de implementación que cuesta aproximadamente 20 USD al mes.
2. Un sencillo caso de uso de un chatbot listo para la producción que se implementa con la configuración predeterminada y se ejecuta en EE. UU. Este (Virginia del Norte), con la tecnología Amazon Bedrock sin acceso a los documentos, lo que también cuesta alrededor de 200 dólares al mes.
3. Un sistema escalado en un caso de uso de Amazon VPC que admite 8000 consultas al día en decenas de miles de documentos, lo que cuesta alrededor de 1500 USD al mes. El coste del caso de uso variará en función de la configuración, por ejemplo, en los casos de uso de texto con distintos proveedores de modelos, con o sin la generación aumentada de recuperación (RAG) activada o no, etc.

Descripción de la carga de trabajo	Coste estimado (USD/mes)
Ejemplo de costo del panel de implementación	20$ al mes
Ejemplo de costos de una prueba de concepto basada en texto (incluye un panel de implementación y un caso de uso de texto, aproximadamente 100 interacciones por día)	40$ al mes
Ejemplo de costos de un motor de consultas generativas de IA altamente escalable (Incluye un panel de implementación, un caso de uso de texto y un índice de Amazon Kendra para RAG de hasta 100 000 documentos con unas 8 000 consultas al día, con VPC habilitad a)	1500$ al mes
Ejemplo de costos de una prueba de concepto basada en un agente (Incluye un panel de implementación, 1 caso de uso de Bedrock Agent con las bases de conocimiento de Amazon Bedrock y Amazon Bedrock Guardrails habilitadas, aproximad amente 100 interacciones por día)	840$ al mes
Ejemplo de costos del servidor MCP (Incluye un panel de implementación, 1 caso de uso de un servidor MCP con el método Gateway para la integración con Lambda, aproximadamente 100 invocaciones de herramientas por día)	22$ al mes
Ejemplos de costos de Agent Builder	55$ al mes

Descripción de la carga de trabajo	Coste estimado (USD/mes)
(Incluye un panel de implementación, 1 caso de uso de Agent Builder con integración MCP y memoria de larga duración habilitada, aproximadamente 100 interacciones por día)	
Ejemplos de costos de Workflow Builder (Incluye un panel de implementación, 1 flujo de trabajo con 3 agentes de Agent Builder y unas 100 interacciones por día)	109$ al mes

> **Important**
>
> Estos ejemplos solo pretenden ayudarlo a estimar los costos de sus cargas de trabajo específicas. El uso de diferentes LLMs configuraciones o servicios de AWS puede cambiar sus costos (por ejemplo, serverless/on-demand billing vs. provisioned/time facturados). Para administrar los costos, recomendamos crear un presupuesto a través de AWS Cost Explorer. Los precios están sujetos a cambios. Para obtener más información, consulte la página web de precios de cada servicio de AWS utilizado en esta solución.

Ejemplo de costos para ejecutar el panel de implementación

En la siguiente tabla se muestra el desglose de los costes de un panel de despliegue con los parámetros predeterminados y 100 usuarios activos en la región de EE. UU. del Este (Virginia del Norte) durante un mes, lo que costará unos 20 dólares al mes.

Servicio de AWS	Dimensiones	Coste [USD]
API Gateway, DynamoDB, CloudFront Amazon S3, Lambda, Systems Manager Parameter Store	5000 llamadas a la API REST de 512 KB al mes sin tener activado el almacenamiento en caché	1,97$

Servicio de AWS	Dimensiones	Coste [USD]
Amazon Cognito	100 usuarios activos al mes con funciones de seguridad avanzadas habilitadas y sin que los usuarios inicien sesión mediante la federación de SAML o OIDC	5,55$
AWS WAF	10 000 solicitudes web en 1 ACL web y 7 reglas definidas sin ningún grupo de reglas	12,60$
Coste total del panel de implementación		20,12$

Ejemplo de costos de una prueba de concepto basada en texto

Un panel de implementación puede tener muchos casos de uso implementados en un momento dado. La siguiente tabla muestra el desglose de los costes de un caso de uso implementado sin RAG para 1 usuario empresarial que realiza 100 consultas al día con el LLM. Las consultas se envían como un mensaje de texto WebSocket y la respuesta se transmite en forma de símbolos, suponiendo que la transmisión esté habilitada. Con el modelo Amazon Bedrock Nova Pro, el costo de ejecutar este caso de uso es de unos 20 dólares al mes.

Servicio de AWS	Dimensiones	Coste [USD]
API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, AWS Systems Manager Parameter Store	100 interacciones de chat por día. Tamaño medio de los mensajes: 32 KB por mensaje y 5 minutos por conexión.	0,61$
CloudWatch	CloudWatch Registros de 1,5 GB con el modo detallado activado para experimentar	7,23$

Servicio de AWS	Dimensiones	Coste [USD]
Amazon DynamoDB	Tabla de historial de conversaciones, 1 GB de almacenamiento Tabla de configuración LLM, 1 GB de almacenamiento	3,05$
Subtotal de los costos de los casos de uso (no incluidos) LLMs		10,89$
Amazon Bedrock (Nova Pro)	Supuestos para 100 interacciones por día: * Coste mensual de 190 000 fichas de entrada al día = 0,152 USD × 30 * Coste mensual de 16 000 fichas de salida al día = 0,0512 × 30 USD	6,10$
Coste total de la aplicación con Amazon Bedrock (Nova Pro)	10,89$ (coste del caso de uso) + 6,10$ (coste de Amazon Bedrock)	17,00$

Note

Los costos de las llamadas de inferencia realizadas a servicios fuera de la red de AWS no se incluyen en estas estimaciones. Consulte la guía de precios de su proveedor de LLM si no utiliza un proveedor de modelos de AWS.
Las guías de precios de los servicios de AWS se encuentran en: precios de Amazon Bedrock y precios de Amazon SageMaker AI.

Ejemplos de costos de un motor de consultas generativas de IA altamente escalable

La siguiente tabla proporciona el desglose de costos de un caso de uso compatible con RAG con el modelo Nova Pro de Amazon Bedrock como LLM. Si se añade una base de conocimientos de Bedrock, este caso de uso cuesta unos 1300$ al mes

Servicio de AWS	Dimensiones	Coste [USD]
API Gateway (WebSocket)	8000 interacciones de chat por día. Tamaño medio de los mensajes: 32 KB por mensaje y 5 minutos por conexión.	38,89$
CloudFront	240 000 solicitudes al mes, con 100 GB de datos transferidos a Internet y 1 GB de datos transferidos al origen	8,76$
Amazon Bedrock (Nova Pro)	Supuestos: Símbolos de entrada = PromptTemplate (400) + contexto (400) + ChatHistory (1080) + símbolos de entrada de consulta (20) = 1900 Tokens de salida = 160 (promedio) Con 8.000 transacciones al día, Coste de los tokens de entrada diarios (1900 x 8000 = 15 200 000 fichas x 0,0008/1000 de precio por ficha)	487,80 DÓLARES

Servicio de AWS	Dimensiones	Coste [USD]
	Coste de los tokens de producción diaria (160 x 8000 = 1,280,000 tokens x 0,0032/1000 de precio por token) Coste mensual ((12,16$ + 4,10$) x 30)	
CloudWatch	24 métricas que utilizan 5 GB de datos ingeridos para los registros y 1 panel	9,72$
DynamoDB	Tabla DynamoDB para realizar un seguimiento del historial de conversaciones con cada registro de hasta 1 KB de datos, 8000 lecturas y escrituras por día	11,70$
Lambda	Tamaño del contenedor: 128 MB, 512 MB efímero almacenamiento, 2 funciones Lambda utilizadas para la autorización Tamaño del contenedor: 256 MB, 512 MB de almacenam iento efímero, 5 solicitudes por segundo con un tiempo de procesamiento promedio de 20 segundos	20,89$
Coste total del caso de uso		577,76$ al mes más el coste de la base de conocimientos (véase más abajo)

> **Note**
>
> Los costos de las llamadas a la API realizadas a cualquier servicio fuera de la red de AWS no se incluyen en estas estimaciones. Consulte la guía de precios de su proveedor de LLM si no utiliza Amazon Bedrock.

Costos de añadir una base de conocimientos

Los costes de la base de conocimientos variarán en función del tipo de base de conocimientos utilizada y (en el caso de Bedrock) del almacén vectorial de apoyo utilizado por la base de conocimientos. El aprovisionamiento y la gestión de las bases de conocimiento están fuera del alcance de la solución.

Bases de conocimiento de Amazon Bedrock

La solución no administra ni aprovisiona ningún recurso relacionado con las bases de conocimiento de Amazon Bedrock. Amazon Bedrock no incurre en costes por el uso de la función de base de conocimientos en sí, pero se le cobrará por el uso del modelo de incrustación utilizado en su caso de uso en cada consulta. Además, el almacén vectorial de respaldo de su base de conocimientos (por ejemplo, un índice de Amazon OpenSearch Service o una base de datos de Amazon Relational Database Service) tendrá un coste asociado que no se puede proporcionar ni calcular aquí.

Para el escenario de motor de consultas de IA generativa de alta escalabilidad anterior, los costos incurridos por este servicio para llamar al modelo de incrustaciones de Amazon Bedrock son los siguientes:

Servicio de AWS	Dimensiones	Coste [USD]
Amazon Bedrock (Amazon Titan Text Embeddings V2)	8000 consultas al día con 1900 fichas de entrada por consulta = 15 200 000 fichas = 0,30 USD al día. Coste diario x 30 días = coste mensual de 9 USD	9,00$

Servicio de AWS	Dimensiones	Coste [USD]
Ejemplo OpenSearch de uso de Amazon Service (Serverless)	Configuración básica sin servidor con 4 unidades de OpenSearch cómputo (OCU) (mínimo facturable) = 23,04 USD por día Coste diario x 30 días = 691,20 USD **Note** Esto proporciona una estimación aproximada, ya que algunas cargas de trabajo requerirán más OCUs, mientras que los clientes con OpenSearch recursos aprovisionados existentes incurrirán en menos costos en este caso.	691,20$
Coste adicional total		700,20$

Amazon Kendra

La solución puede proporcionarle un índice de Kendra o puede traer el suyo propio. El costo de ejecutar una configuración adecuada para el motor de consultas generativas de IA altamente escalable mencionado anteriormente es el siguiente:

Servicio de AWS	Dimensiones	Coste [USD]
Amazon Kendra	De 0 a 8 000 consultas al día y hasta 100 000 documentos con Amazon Kendra Enterprise Edition con 0 a 50 fuentes de datos	1.008,00$

> **Note**
>
> Puede compartir el índice de Amazon Kendra entre casos de uso, pero esto puede aumentar el número de consultas por índice. Si no está incluido en la edición Amazon Kendra Enterprise, se aplicarán cargos adicionales.

Coste incremental de habilitar Amazon VPC para un caso de uso

La siguiente tabla proporciona el desglose de los costos de habilitar Amazon VPC para un caso de uso implementado en dos. AZs

Servicio de AWS	Dimensiones	Coste [USD]
Puerta de enlace Amazon NAT	Supuesto: implementación de 2 zonas de disponibilidad, con una puerta de enlace NAT en cada zona de disponibilidad. 100 GB de datos procesados a través de NAT Gateway durante 730 horas, 100 GB de datos procesados por mes	74,70$
AWS PrivateLink (puntos de enlace de VPC)	Supuestos: implementación de 2 zonas de disponibilidad, con 1 subred privada en cada zona de disponibilidad y 1 punto final de VPC	97,84$

Servicio de AWS	Dimensiones	Coste [USD]
	con 2 interfaces ENIs de red elásticas (). 6 puntos de enlace de VPC, 2 por punto de enlace de ENIs VPC, 730 horas con 1024 GB de datos procesados en un mes	
Dirección pública IPv4	Supuesto: implementación de 2 zonas de disponibilidad, 1 subred pública en cada zona de disponibilidad con una puerta de enlace NAT en cada subred pública. Cada puerta de enlace NAT está configura da con 1 puerta pública activa. IPv4 2 IPv4 direcciones públicas activas x 730 horas en un mes x 0,005$ por hora = 7,3 USD	7,30$
costos adicionales (para Amazon VPC)		179,93 DÓLARES

Implicaciones en materia de costos al utilizar el rendimiento aprovisionado

Los costes del rendimiento aprovisionado variarán en función del tipo de modelo que haya aprovisionado y del período de compromiso, así como de las unidades modelo seleccionadas para el período de compromiso. El uso del rendimiento aprovisionado conlleva un coste adicional.

Para obtener más información y obtener la mayor cantidad up-to-date de precios, consulta los precios de Bedrock.

Coste del uso de la inferencia entre regiones

El uso de la inferencia entre regiones no conlleva ningún coste adicional de enrutamiento o transferencia de datos. Pagas el mismo precio por token por los modelos que en tu región de origen o principal.

Muestra los costes de una prueba de concepto basada en un agente

Cuando utiliza Amazon Bedrock Agents, se le cobra en función de los componentes que componen el agente, como el modelo de respaldo y la base de conocimientos (si RAG está habilitado), además de las capacidades adicionales que añada. La siguiente tabla muestra el desglose de costos de un caso de uso de Bedrock Agent configurado con un modelo Claude 3.5 Sonnet bajo demanda, Amazon Bedrock Knowledge Bases y Amazon Bedrock Guardrails.

Al igual que el costo de añadir las bases de conocimiento de Amazon Bedrock, esta solución no administra ni aprovisiona los recursos relacionados con los agentes de Amazon Bedrock. La solución tampoco implica costes por el uso de las bases de conocimiento de Amazon Bedrock, pero sí los costes de:

- Utilizar el modelo de incrustación para cada consulta que se le envíe
- El almacén de vectores de respaldo de su base de conocimientos (por ejemplo, un índice de Amazon OpenSearch Service o una base de datos de Amazon RDS)

En la siguiente tabla se presuponen 100 interacciones por día con 1900 tokens de entrada y 160 tokens de salida por consulta.

> **Note**
>
> En este ejemplo de caso de uso de Bedrock Agent, si hubiera un grupo de acción configurado para usar una API externa, esos costos serían adicionales. Están fuera del ámbito de los cálculos de esta tabla.

Servicio de AWS	Dimensiones	Coste [USD]
API Gateway (WebSocket) CloudFront, Lambda, Amazon	100 interacciones de chat al día, tamaño medio de los	0,61$

Servicio de AWS	Dimensiones	Coste [USD]
S3, almacén de parámetros de Systems Manager	mensajes: 32 KB por mensaje, 5 minutos por conexión	
CloudWatch	CloudWatch Registros de 1,5 GB con el modo detallado activado para experimentar	7,23$
DynamoDB	Tabla de configuración LLM para un tamaño de registro de 1 KB y 1 GB de almacenam iento	0,25$
Subtotal de los costos (no incluidos) LLMs		8,09$
Soneto antrópico Claude 3.5	* El coste diario de 190 000 fichas de entrada al día (0,003/1000 fichas) = 0,57$ + Coste diario × 30 días = 17,10$ * Coste diario de 16 000 fichas de salida por día (0,015 €/1 000 fichas) = 0,24$ + Coste diario × 30 días = 7,20$	24,30$
Amazon Bedrock (Amazon Titan Text Embeddings V2) para las bases de conocimie nto de Amazon Bedrock	Coste diario de 190 000 fichas de entrada al día (0,00002/ 1000 fichas) = 0,004 Coste diario × 30 días = 0,12$	0,12$

Servicio de AWS	Dimensiones	Coste [USD]
Ejemplo OpenSearch de uso de Amazon Service (Serverless)	Configuración básica sin servidor con 4 unidades de OpenSearch cómputo (OCU) (mínimo facturable) = 23,04$ por día Coste diario × 30 días = 691,20$	691,20$
Barreras de protección para Amazon Bedrock	190 000 fichas equivalen aproximadamente a 760 000 (190 000 × 4) caracteres y 3 800 unidades de texto (760 000 caracteres/200) Pensemos en una barandilla configurada con filtros de contenido, filtro de información de identificación personal (PII), filtro de información confidencial (expresión regular) y filtros de palabras Coste diario del filtro de contenido (0,75 €/1000 unidades de texto) más coste del filtro de PII (0,1/1000 unidades de texto) + filtro de información confidencial (expresiones regulares) + filtros de palabras = 2,85$ + 0,38$ + 0$ Coste mensual = coste diario × 30 días = 96,90$	96,90$

Servicio de AWS	Dimensiones	Coste [USD]
Coste total de la solicitud de un agente respaldado por Anthropic Claude 3.5 Sonnet	8,09$ (coste por caso de uso) + 812,52$ (otras configura ciones de agentes)	820,61$

> **Note**
>
> Consulte la guía de precios de su proveedor de LLM si no utiliza un proveedor de modelos de AWS. Las guías de precios de los servicios de AWS se encuentran en: precios de Amazon Bedrock y precios de Amazon SageMaker AI.

Ejemplos de costos del servidor MCP

Los casos de uso del servidor MCP permiten la implementación y la administración de servidores del Model Context Protocol en Amazon Bedrock AgentCore. La siguiente tabla muestra el desglose de costos de un caso de uso de un servidor MCP que utiliza el método Gateway para empaquetar las funciones Lambda existentes.

La solución administra la implementación y la AgentCore configuración de Gateway. Se le cobrará por:

- Costes de infraestructura (API Gateway, Lambda, DynamoDB, S3) CloudWatch
- AgentCore Consumo de gateway (por invocación de herramienta)
- Costes de ejecución de la función Lambda (para el método Gateway con objetivos Lambda)
- Costes de API externas (para el método Gateway con destinos de API o servidor MCP, si corresponde)

Elemento	Cálculos	Costo
Amazon API Gateway (API REST)	100 invocaciones de herramientas por día × 30 días = 3000 solicitudes al mes	0,05 USD

Elemento	Cálculos	Costo
AWS Lambda (orquestación)	100 invocaciones al día × 30 días × 1 segundo de media × 512 MB = 3000 GB-segundos al mes	0,05 USD
Amazon DynamoDB	3000 read/write solicitudes al mes más 1 GB de almacenamiento	0,15$
Amazon CloudWatch	Supervisión y registro estándar para 3000 invocaciones	1,00$
Amazon S3	Almacenamiento de configuración y registros (uso mínimo)	0,25$
Amazon Bedrock Gateway AgentCore	3000 invocaciones de herramientas al mes	0,05 USD
Función Lambda objetivo	100 invocaciones por día × 30 días × 0,5 segundos × 128 MB = 1500 GB-segundos por mes	0,25$
Coste mensual total	1,75$ (infraestructura) + 0,05$ (puerta de enlace) AgentCore	1,80$

> **Note**
>
> Los costos varían según el método de implementación (Gateway o Runtime), los tipos de objetivos y los patrones de uso. Las implementaciones del método Runtime incurren en cargos AgentCore de tiempo de ejecución en lugar de cargos de Gateway. Los costos de las API externas y los costos de alojamiento de contenedores personalizados son adicionales.

Ejemplo de costos de Agent Builder

Agent Builder le permite crear e implementar agentes personalizados en Amazon Bedrock AgentCore. La siguiente tabla muestra el desglose de costos de un caso de uso de Agent Builder configurado con Claude 3.5 Sonnet, integración de servidores MCP y memoria de larga duración habilitada.

La solución gestiona el despliegue y la configuración del AgentCore entorno de ejecución. Se le cobrará por:

- Costes de infraestructura (API Gateway, Lambda, DynamoDB, S3) CloudWatch
- AgentCore Consumo de tiempo de ejecución (horas de CPU y memoria en función del tiempo real de ejecución del agente)
- Inferencia del modelo básico (fichas de entrada y salida)
- AgentCore Memoria (eventos a corto plazo y almacenamiento/recuperación a largo plazo)

En la siguiente tabla se presuponen 100 interacciones por día con 1900 símbolos de entrada y 160 símbolos de salida por consulta, con un tiempo medio de ejecución del agente de 5 segundos por interacción.

Servicio de AWS	Dimensiones	Coste [USD]
API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, almacén de parámetros de Systems Manager	100 interacciones de chat al día, tamaño medio de los mensajes: 32 KB por mensaje, 5 minutos por conexión	0,61$
CloudWatch	CloudWatch Registros de 1,5 GB con el modo detallado activado para experimentar	7,23$
DynamoDB	Tabla de configuración LLM para un tamaño de registro de 1 KB y 1 GB de almacenam iento	0,25$

Servicio de AWS	Dimensiones	Coste [USD]
Subtotal de los costos de infraestructura		8,09$
Tiempo de ejecución de Amazon Bedrock AgentCore	* CPU: 1 vCPU × 5 segundos × 100 interacciones = 125 vCPU- seconds/day = 0.140 vCPU-hours/day + Coste diario: 0,140 × 0,0895$ = 0,013$ + Coste mensual: 0,013 × 30$ = 0,38$ * Memoria: 512 MB (0,5 GB) × 5 segundos × 100 interacci ones = 250 GB- seconds/day = 0.069 GB-hours/day Más coste diario: 0,069 × 0,00945$ = 0,0007$ + Coste mensual: 0,0007$ × 30 = 0,02$	0,40$
Soneto antrópico Claude 3.5	* El coste diario de 190 000 fichas de entrada al día (0,003/1000 fichas) = 0,57$ + coste diario × 30 días = 17,10$ * El coste diario de 16 000 fichas de salida al día (0,015 000 fichas) = 0,24$ + coste diario × 30 días = 7,20$	24,30$

Servicio de AWS	Dimensiones	Coste [USD]
Memoria Amazon Bedrock AgentCore	* Memoria a corto plazo: 100 eventos nuevos events/day × 0,25/1000 $ = 0,025$ al día + Coste mensual: 0,025$ × 30$ = 0,75$ * Almacenamiento de memoria a largo plazo (estrategia integrada): 100 registros × 0,75 dólares/1 000$ = 0,075$ al mes records/month * Recuperación de memoria a largo plazo: 100 retrievals/day × 0,50$ cada 1000 recuperaciones = 0,05$ al día + Coste mensual: 0,05$ × 30$ = 1,50$	2,33$
Coste total de la aplicación de Agent Builder con Claude 3.5 Sonnet	8,09$ (infraestructura) + 0,40$ (tiempo de AgentCore ejecución) + 24,30$ (modelo) + 2,33$ (memoria)	35,12$

Note

AgentCore Los precios del tiempo de ejecución se basan en el consumo. Los costos reales dependen de:

- Tiempo de ejecución del agente (uso de CPU y memoria durante el procesamiento activo)
- Número de interacciones y su complejidad
- Uso de la herramienta MCP (adicional CPU/memory para la ejecución de la herramienta)
- Configuración de memoria (habilitada para memoria a corto plazo o memoria a largo plazo)

Para ver AgentCore los precios detallados, consulta los precios de Amazon Bedrock.

> **Note**
>
> Si utiliza servidores MCP que utilizan servicios APIs o servicios externos, esos costos son adicionales y están fuera del ámbito de este cálculo. Del mismo modo, si utiliza herramientas de AgentCore navegador o interpretación de código, se aplican cargos basados en el consumo: 0,0895 USD por hora de CPU virtual y 0,00945 USD por GB-hora.

Ejemplo de costos de Workflow Builder

Workflow Builder crea un agente supervisor que organiza varios agentes de Agent Builder. La siguiente tabla muestra el desglose de costos de un flujo de trabajo con 1 agente supervisor y 3 agentes especializados de Agent Builder, todos configurados con Claude 3.5 Sonnet y con memoria de larga duración habilitada.

Supuestos: 100 interacciones por día, media de 2 delegaciones de agentes por interacción, tiempo de ejecución de 5 segundos por agente.

Servicio de AWS	Dimensiones	Coste [USD]
API Gateway (WebSocket) CloudFront, Lambda, Amazon S3, almacén de parámetros de Systems Manager	100 interacciones de chat al día, tamaño medio de los mensajes: 32 KB por mensaje, 5 minutos por conexión	0,61$
CloudWatch	CloudWatch Registros de 1,5 GB con el modo detallado activado para experimentar	7,23$
DynamoDB	Tabla de configuración LLM para un tamaño de registro de 1 KB y 1 GB de almacenam iento	0,25$

Servicio de AWS	Dimensiones	Coste [USD]
Subtotal de los costos de infraestructura		8,09$
Amazon Bedrock AgentCore Runtime (agente supervisor)	* CPU: 1 vCPU × 5 segundos × 100 interacciones = 0,140 vCPU hours/day × 30 = $0.38 * Memory: 0.5 GB × 5 seconds × 100 interactions = 0.069 GB-hours/day - × 30 = 0,02$	0,40$
Amazon Bedrock AgentCore Runtime (3 agentes especializados)	* Un promedio de 2 delegaciones por interacción = 200 agentes executions/day * CPU: 1 vCPU × 5 seconds × 200 = 0.278 vCPU-hours/day × 30 = $0.75 * Memory: 0.5 GB × 5 seconds × 200 = 0.139 GB-hours/day × 30 = 0,04$	0,79$
Anthropic Claude 3.5 Sonnet (agente supervisor)	* Entrada: 190 000 × 0,003/1 000$ = 0,57 dólares/día tokens/day × 30 = 17,10$ * Salida: 16 000 × 0,015 dólares/1 K = 0,24 dólares/día × 30 = 7,20$ tokens/day	24,30$
Anthropic Claude 3.5 Sonnet (agentes especializados)	* Media de 2 delegaciones por interacción * Entrada: 380 000 tokens/day × 0,003$ = 1,14 dólares/día × 30 = 34,20$ * Salida: 32 000 × 0,015 dólares/1 000$ = 0,48 dólares/día × 30$ = 14,40$ tokens/day	48,60$

Servicio de AWS	Dimensiones	Coste [USD]
Amazon Bedrock AgentCore Memory (agente supervisor)	* A corto plazo: 100 events/day × 0,25/1 000$ × 30 = 0,75$ * Almacenamiento a largo plazo: 100 registros × 0,75 dólares/1 000$ = 0,08$ * Recuperación a largo plazo: 100 × 0,50 €/1000 × 30$ = 1,50$ retrievals/day	2,33$
Amazon Bedrock AgentCore Memory (agentes especializados)	* A corto plazo: 200 events/day × 0,25/1 000$ × 30 = 1,50$ * Almacenamiento a largo plazo: 200 registros × 0,75 dólares/1 000$ = 0,15$ * Recuperación a largo plazo: 200 × 0,50 €/1000 × 30$ = 3,00$ retrievals/day	4,65$
Coste total de la aplicación para Workflow Builder con 3 agentes	8,09$ (infraestructura) + 1,19$ (AgentCore tiempo de ejecución) + 72,90$ (modelos) + 6,98$ (memoria)	89,16$

Note

- Las tasas de delegación más altas aumentan proporcionalmente el consumo de fichas

Para ver AgentCore los precios detallados, consulta los precios de Amazon Bedrock.

Seguridad

Cuando crea sistemas en la infraestructura de AWS, las responsabilidades de seguridad se comparten entre usted y AWS. Este modelo de responsabilidad compartida reduce la carga

operativa, ya que AWS opera, administra y controla los componentes, incluidos el sistema operativo anfitrión, la capa de virtualización y la seguridad física de las instalaciones en las que operan los servicios. Para obtener más información sobre la seguridad de AWS, visite Seguridad en la nube de AWS.

Uso de modelos de cimentación en Amazon Bedrock

Amazon Bedrock alberga una colección de modelos, desde modelos Amazon Nova hasta otros modelos de bases líderes (FMs). Cuando se utiliza Amazon Bedrock, todos los modelos se alojan en la infraestructura de AWS. Esto significa que cuando utilice Amazon Bedrock como proveedor de LLM, todas sus solicitudes de inferencia permanecerán en la red de AWS y el tráfico de red no saldrá de su región.

> **Note**
>
> Todos los modelos básicos (FMs) disponibles a través de Amazon Bedrock se alojan directamente en la infraestructura de AWS gestionada y propiedad de AWS. Los proveedores de modelos no tienen acceso a los datos de los clientes, como las indicaciones y las continuaciones, ni a los registros de servicio de Amazon Bedrock. Para obtener información adicional sobre la postura de seguridad de Amazon Bedrock, consulte Protección de datos en Amazon Bedrock en la Guía del usuario de Amazon Bedrock.

Roles de IAM

Las funciones de IAM permiten a los clientes asignar políticas y permisos de acceso detallados a los servicios y usuarios de la nube de AWS. Esta solución crea funciones de IAM que otorgan acceso a las funciones Lambda de la solución para crear recursos regionales.

CloudWatch Registros

Puede habilitar el modo detallado al implementar un caso de uso en la página de selección del modelo del panel de implementación, en la sección Configuración adicional. El modo detallado permite realizar CloudWatch registros detallados que pueden ser útiles para la depuración y acelerar la experimentación.

> **Note**
>
> Cuando el modo detallado está activado, también se registran los documentos recuperados de la base de conocimientos (si el RAG está activado) y las solicitudes, que pueden contener información confidencial.

VPC

La solución ofrece dos opciones para la configuración de Amazon VPC:

1. Deje que la solución cree una Amazon VPC para usted.
2. Administrar y traer su propia Amazon VPC para usarla dentro de la solución.

Deje que la solución cree una Amazon VPC para usted

Si selecciona la opción de permitir que la solución cree una Amazon VPC, se implementará como una arquitectura 2-AZ de forma predeterminada con un rango de CIDR de 10.10.0.0/20. Tiene la opción de usar Amazon VPC IP Address Manager (IPAM), con 1 subred pública y 1 subred privada en cada zona de disponibilidad. La solución crea pasarelas NAT en cada una de las subredes públicas y configura las funciones Lambda para crearlas en ENIslas subredes privadas. Además, esta configuración crea tablas de enrutamiento y sus entradas, grupos de seguridad y sus reglas ACLs, redes y puntos finales de VPC (puntos de enlace e interfaz).

Administrar su propia Amazon VPC

Al implementar la solución con una Amazon VPC, tiene la opción de usar una Amazon VPC existente en su cuenta y región de AWS. Le recomendamos que ponga su VPC a disposición en al menos dos zonas de disponibilidad para garantizar una alta disponibilidad. Su VPC también debe tener los siguientes puntos de enlace de VPC y sus políticas de IAM asociadas para las configuraciones de VPC y tabla de enrutamiento.

Para un panel de implementación, Amazon VPC

1. Punto final de puerta de enlace para DynamoDB.
2. Punto final de puerta de enlace para S3.
3. Punto final de interfaz para CloudWatch.

4. Punto final de interfaz para AWS CloudFormation.

Para un caso de uso Amazon VPC

1. Punto final de puerta de enlace para DynamoDB.
2. Punto final de puerta de enlace para S3.
3. Punto final de interfaz para CloudWatch.
4. Punto final de interfaz para el almacén de parámetros de Systems Manager.

> **Note**
>
> La solución solo requiere`com.amazonaws.region.ssm`.

5. Punto final de interfaz para Amazon Bedrock (bedrock-runtime, agent-runtime,). bedrock-agent-runtime
6. Opcional: si la implementación utilizará Amazon Kendra como base de conocimientos, se necesitará un punto de enlace de interfaz para Amazon Kendra.
7. Opcional: si la implementación utilizará cualquier LLM en Amazon Bedrock, se necesitará un punto de enlace de interfaz para Amazon Bedrock.

> **Note**
>
> La solución solo requiere. `com.amazonaws.region.bedrock-runtime`

8. Opcional: si la implementación utilizará Amazon SageMaker AI para la LLM, se necesitará un punto final de interfaz para Amazon SageMaker AI.

> **Note**
>
> La solución no eliminará ni modificará la configuración de la VPC al utilizar la opción de implementación Bring your own VPC. Sin embargo, eliminará todas las VPCs que haya creado la solución en la opción Crear una VPC para mí. Por este motivo, debe tener cuidado al compartir una VPC gestionada por una solución entre pilas o despliegues.
> Por ejemplo, la implementación A usa la opción Crear una VPC para mí. La implementación B usa Bring my own VPC mediante la VPC creada por la implementación A. Si la implementación A se elimina antes que la implementación B, la implementación B dejará

de funcionar porque se eliminó la VPC. Además, dado que la implementación B utiliza las funciones ENIs creadas por Lambda, la eliminación de la implementación A puede provocar errores y retener los recursos residuales.

Amazon CloudFront

Esta solución implementa una consola web alojada en un bucket de Amazon S3. Para ayudar a reducir la latencia y mejorar la seguridad, esta solución incluye una CloudFront distribución con una identidad de acceso de origen, es decir, un CloudFront usuario que proporciona acceso público al contenido del bucket del sitio web de la solución. Para obtener más información, consulte Restringir el acceso al contenido de Amazon S3 mediante una identidad de acceso de origen en la Guía para CloudFront desarrolladores de Amazon.

> **Note**
>
> CloudFront tiene un límite de cuota flexible a nivel de cuenta de 20 políticas de encabezados de respuesta. Esta solución crea políticas de encabezados de respuesta personalizadas por motivos de seguridad. Si tiene más de 20 implementaciones del Generative AI Application Builder en AWS o sus casos de uso, es posible que las nuevas implementaciones fallen por alcanzar el límite de cuota.

Para resolver este problema, puede solicitar un aumento de cuota para la cuota de Response Header Policies en la consola de AWS Service Quotas siguiendo estos pasos:

1. Abra la consola de AWS Service Quotas.
2. En el panel de navegación, seleccione Servicios de AWS.
3. Busca y selecciona Amazon CloudFront.
4. Ve a la cuota de políticas del encabezado de respuesta y selecciona Solicitar aumento de cuota.
5. Siga las instrucciones para solicitar un aumento del límite de cuota de su cuenta de AWS.

Al aumentar la cuota de políticas de cabecera de respuesta, puede asegurarse de que las nuevas implementaciones del Generative AI Application Builder en AWS o sus casos de uso no fallen debido al límite de cuota.

Cuotas

Las cuotas de servicio (que también se denominan límites) establecen el número máximo de recursos u operaciones de servicio para su cuenta de AWS.

Cuotas para servicios de AWS en esta solución

Asegúrese de tener una cuota suficiente para cada uno de los servicios implementados en esta solución. Para obtener más información, consulte las cuotas de servicio de AWS.

Use los enlaces siguientes para ir a la página de ese servicio. Para ver las cuotas de servicio para todos los servicios de AWS en la documentación sin cambiar de página, consulte la información de la página de Cuotas y puntos de conexión del servicio del PDF.

Cuotas de Amazon Bedrock AgentCore

Para las implementaciones de Agent Builder, tenga en cuenta las siguientes cuotas de AgentCore servicio de Amazon Bedrock:

Cuota	Este de EE. UU. (Norte de Virginia)	Otras regiones
Cargas de trabajo de sesión activas por cuenta	1 000	500
Total de agentes por cuenta	1 000	1 000
Versiones por cuenta	1 000	1 000

Implementación de la solución

Esta solución utiliza CloudFormation plantillas y pilas de AWS para automatizar su implementación. La CloudFormation plantilla especifica los recursos de AWS incluidos en esta solución y sus propiedades. La CloudFormation pila aprovisiona los recursos que se describen en la plantilla.

Información general del proceso de implementación

Antes de lanzar la solución, revise el costo, la arquitectura, la seguridad y otras consideraciones que se describen en esta guía.

> **Important**
>
> Si planea usar Amazon Bedrock, debe solicitar acceso a los modelos antes de que estén disponibles para su uso. Consulte Model access en la Guía del usuario de Amazon Bedrock para obtener más información.

Tiempo de implementación: aproximadamente 10 minutos

Paso 1: Inicie la pila de paneles de implementación

Paso 2: Implementar un caso de uso

Paso 3: Implemente un caso de uso mediante el asistente del panel de implementación

Paso 4: Configuración posterior a la implementación

Opcionalmente, puede implementar los casos de uso por separado de la solución, si prefiere no tener la interfaz de usuario del panel de implementación o APIs.

- Implementación de un caso de uso de Text independiente
- Implementación de un caso de uso de Bedrock Agent independiente

También puede proporcionar una configuración de chat de DynamoDB.

> **Important**
>
> Esta solución envía métricas operativas a AWS (los «datos») sobre el uso de esta solución. Utilizamos estos datos para comprender mejor cómo utilizan los clientes esta solución y los servicios y productos relacionados. La recopilación de estos datos por parte de AWS está sujeta a la Política de privacidad de AWS.

CloudFormation Plantilla de AWS

Puede descargar la CloudFormation plantilla de esta solución antes de implementarla.

View template

genera ai-application-builder-on-aws.template: utilice esta plantilla para lanzar la solución y todos los componentes asociados. La configuración predeterminada implementa las soluciones principales y de soporte que se encuentran en los servicios de AWS en esta sección de soluciones, pero puede personalizar la plantilla para que se adapte a sus necesidades específicas.

> **Note**
>
> Los CloudFormation recursos de AWS se crean a partir de componentes del AWS Cloud Development Kit (AWS CDK).

Esta CloudFormation plantilla de AWS implementa Generative AI Application Builder en AWS en la nube de AWS.

Paso 1: Ejecute la pila de paneles de implementación

Siga las step-by-step instrucciones de esta sección para configurar e implementar la solución en su cuenta.

Tiempo de implementación: aproximadamente 10 minutos

1. Inicie sesión en la consola de administración de AWS y seleccione el botón para lanzar la `generative-ai-application-builder-on-aws.template` CloudFormation plantilla.

2. La plantilla se lanza en la región Este de EE. UU. (Norte de Virginia) de forma predeterminada. Para lanzar la solución en una región de AWS diferente, utilice el selector de regiones de la barra de navegación de la consola.

   > **Note**
   >
   > Esta solución utiliza Amazon Kendra y Amazon Bedrock, que actualmente no están disponibles en todas las regiones de AWS. Si utiliza estas funciones, debe lanzar esta solución en una región de AWS en la que estén disponibles estos servicios. Para obtener la disponibilidad más reciente por región, consulte la lista de servicios regionales de AWS.

3. En la página Crear pila, verifique que la dirección URL de la plantilla correcta se encuentre en el cuadro de texto URL de Amazon S3 y elija Siguiente.
4. En la página Especificar los detalles de la pila, especifique un nombre para la pila. Para obtener información sobre las limitaciones de nombres de caracteres, consulte los límites de IAM y STS en la Guía del usuario de AWS Identity and Access Management.
5. En Parámetros, revise los parámetros de esta plantilla de solución y modifíquelos según sea necesario. Esta solución utiliza los siguientes valores predeterminados.

Parámetro	Predeterminado	Description (Descripción)
Correo electrónico del usuario administrador	No	La dirección de correo electrónico del usuario administrador que tendrá acceso al panel de implement ación. Si se proporcionan, se crearán un grupo y un usuario de Amazon Cognito con permisos para implement ar y gestionar casos de uso. También puede utilizarlos `placeholder@exampl`

Parámetro	Predeterminado	Description (Descripción)
		`e.com` para crear el grupo, pero no el usuario. Consulte la configuración manual del grupo de usuarios para obtener información sobre cómo configurar el grupo de usuarios.
VpcEnabled	No	¿Debería implementarse el panel de implementación en una VPC?
CreateNewVpc	No	Solo está disponible, si lo VpcEnabledestá`Yes`. Si el valor es`Yes`, la pila creará la VPC e implementará la solución dentro de la VPC creada. Si VpcEnabledes `Yes` y CreateNewVpcesNo, debe proporcionar una configura ción de VPC existente (ExistingVpcId,, ExistingP rivateSubnetIdsExistingS ecurityGroupIds, VpcAzs).
IPAMPoolId	(Entrada opcional)	Puede configurar el IPAM y proporcionar el identificador creado como entrada para asignar el rango de direccion es IP que debe utilizar la implementación de esta pila. Para obtener más informaci ón sobre el IPAM, consulte Cómo funciona el IPAM.

Parámetro	Predeterminado	Description (Descripción)
Implemente la interfaz	`Yes`	Tiene la opción de implement ar el panel de implement ación sin la interfaz de usuario web (y sin los recursos de AWS necesarios para la implementación web). En ese caso, la solución implementará toda la infraestr uctura, incluidos los puntos de enlace de la API REST. Esta opción resulta útil para integrar su propia interfaz web con el panel APIs de implementación.
ExistingVpcId	(Entrada opcional)	Necesario solo si desea implementar la solución en una VPC existente que haya creado.
ExistingPrivateSubnetIds	(Entrada opcional)	Necesario solo si desea implementar la solución en una VPC existente que haya creado. Las funciones de Lambda se implementarán en esta subred.
ExistingSecurityGroupIds	(Entrada opcional)	Necesario solo si desea implementar la solución en una VPC existente que haya creado. Asegúrese de que los grupos de seguridad tengan los permisos para una conexión TCP saliente.

Parámetro	Predeterminado	Description (Descripción)
VpcAzs	(Entrada opcional)	Necesario solo si desea implementar la solución en una VPC existente que haya creado.
CognitoDomainPrefix	(Entrada opcional)	Necesario solo si desea implementar la solución en un grupo de usuarios de Amazon Cognito existente que haya creado. Si no proporciona un valor, la solución lo genera.
ExistingCognitoUserPoolId	(Entrada opcional)	Necesario solo si desea implementar la solución en un grupo de usuarios de Amazon Cognito existente que haya creado.
ExistingCognitoUserPoolClient	(Entrada opcional)	Necesario solo si desea implementar la solución en un grupo de usuarios de Amazon Cognito existente que haya creado. Si no proporciona un valor, la solución crea un cliente de grupo de usuarios. Este parámetro solo se puede proporcionar si se proporciona un ExistingCognitoUserPoolIdvalor.

6. Elija Siguiente.
7. En la página Configurar opciones de pila, elija Siguiente.

8. En la página Revisar y crear, revise y confirme la configuración. Seleccione la casilla para confirmar que la plantilla creará los recursos de AWS Identity and Access Management (IAM).
9. Elija Crear para implementar la pila.

Puede ver el estado de la pila en la CloudFormation consola de AWS en la columna Estado. Debería recibir el estado CREATE_COMPLETE en aproximadamente 10 minutos.

Paso 2: implementar un caso de uso

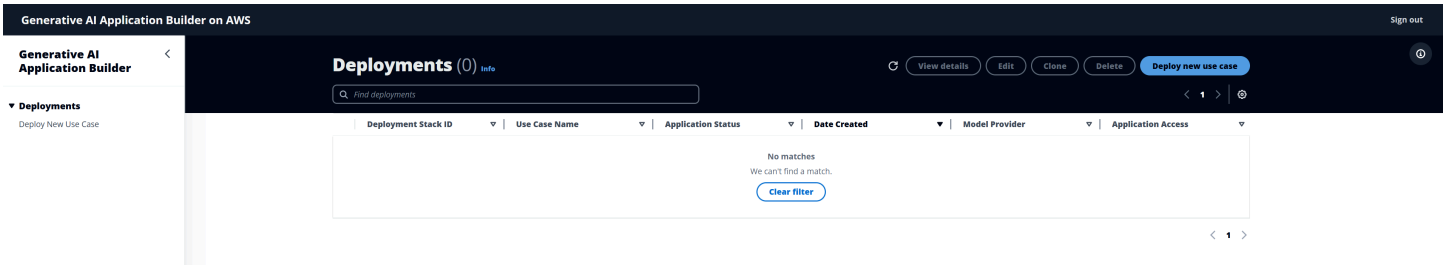
> **Important**
>
> Una vez que la pila se haya implementado correctamente, se envía un correo electrónico de registro al correo electrónico del usuario administrador configurado. Con esas credenciales, el usuario administrador puede iniciar sesión en el panel de implementación para usar la aplicación web.

> **Note**
>
> El DevOps usuario con acceso a la consola de administración de AWS debe proporcionar al usuario administrador la CloudFront URL de la interfaz de usuario del panel de implementación cuando se complete la pila. La URL se encuentra en la pestaña Resultados de la CloudFormation pila.

1. Inicie sesión en el panel de implementación como usuario administrador.
2. En la página de inicio de la aplicación, selecciona Implementar un nuevo caso de uso.

   Se abrirá el asistente de despliegue, que le guiará por el proceso de creación del caso de uso.

Muestra la página de inicio del panel de implementación: implementación nueva



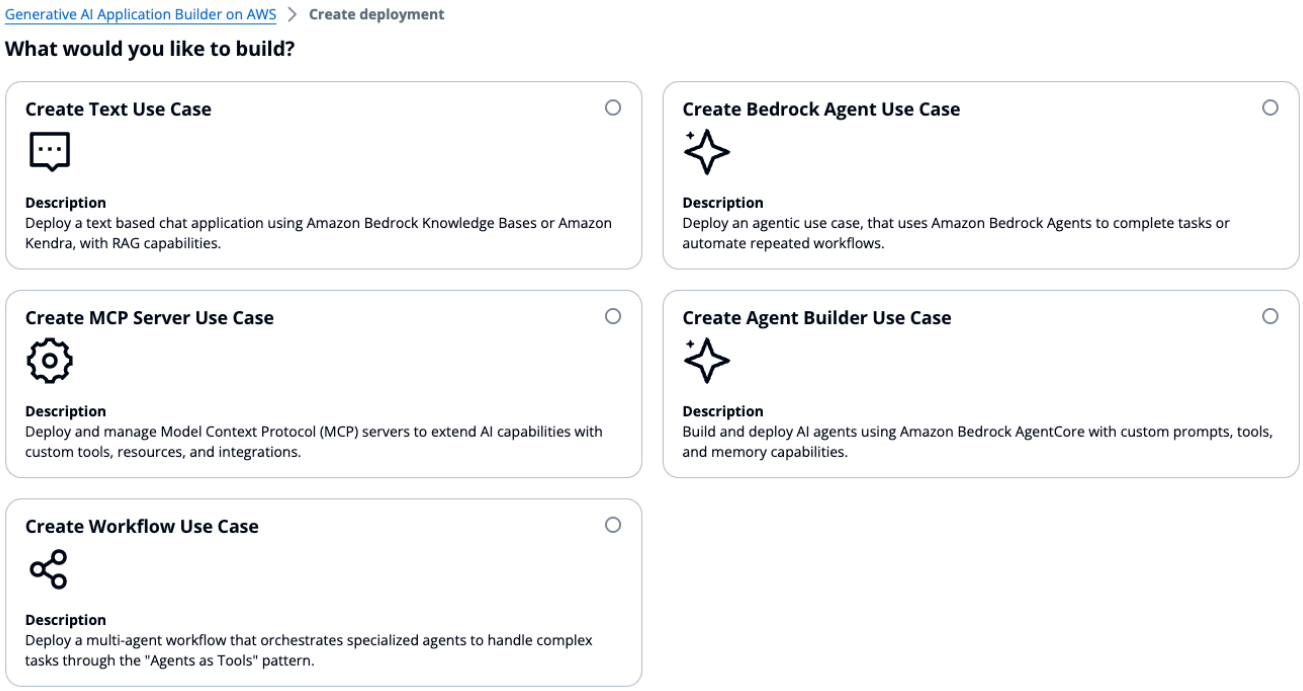
> **Note**
>
> Si necesita añadir usuarios adicionales a su implementación, consulte Administración del grupo de usuarios de Cognito para obtener más información.

Paso 3: Implemente un caso de uso mediante el asistente del panel de implementación

En el asistente del panel de implementación, debe elegir entre las siguientes opciones:

- Caso de uso de texto: despliega una aplicación de chat con funciones RAG opcionales
- Caso de uso de Bedrock Agent: utiliza Amazon Bedrock Agents para completar tareas o automatizar flujos de trabajo repetidos
- Servidor MCP: implemente y administre servidores MCP con métodos de puerta de enlace o tiempo de ejecución
- Agent Builder: cree e implemente agentes personalizados AgentCore con la integración de MCP y la administración de memoria
- Generador de flujos de trabajo: Organice varios agentes de Agent Builder mediante la delegación jerárquica

Muestra cinco opciones: crear un caso de uso de texto, crear un caso de uso de Bedrock Agent, crear un caso de uso de MCP Server, crear un caso de uso de Agent Builder o crear un caso de uso de Workflow.



Paso 3a: implementar un caso de uso de Text

En esta sección se proporcionan instrucciones para implementar un caso de uso de Text.

Seleccione un caso de uso

Al elegir Crear caso de uso de texto, la interfaz de usuario abre la pantalla Seleccionar caso de uso. Proporcione la información siguiente:

- Nombre del caso de uso.
- Dirección de correo electrónico opcional para que el usuario predeterminado del caso de uso se añada al grupo de usuarios de Amazon Cognito para el caso de uso y se le concedan permisos para interactuar con él.
- Si desea implementar una interfaz de usuario con este caso de uso. Si no quieres implementar una interfaz de usuario con el caso de uso, puedes usar los puntos finales de la API implementados para usarlos con tu aplicación.

Detalles del caso de uso

El paso de detalles del caso de uso le permite configurar ajustes adicionales para su implementación.

De forma predeterminada, el caso de uso de Text crea y configura un grupo de usuarios de Amazon Cognito cuando la solución implementa el panel de implementación. La solución autentica los nuevos casos de uso con un cliente recién creado en el mismo grupo de usuarios. Sin embargo, puede proporcionar un ID de grupo de usuarios y un ID de cliente existentes en este paso si quiere usar su propio grupo de usuarios y cliente de Amazon Cognito con este caso de uso.

Important

Los usuarios administradores tienen acceso a todos los casos de uso implementados cuando se crea el grupo de usuarios de Amazon Cognito mediante el asistente de implementación. Si proporciona su propio grupo de usuarios durante la implementación, debe asegurarse de que el administrador tenga los permisos para acceder a los casos de uso implementados. También tendrás que actualizar la devolución de llamada permitida URLs y el cierre de sesión permitido URLs en tus clientes de aplicaciones en Cognito. Para ello:

1. Navegue hasta la consola de Cognito
2. Elija Grupos de usuarios.
3. Elija su grupo de usuarios.
4. Selecciona App Clients en el menú de la izquierda.
5. Elija el cliente de aplicaciones que desee modificar.
6. Seleccione la pestaña Páginas de inicio de sesión.
7. Selecciona Editar y añade tu URLs.
8. Seleccione Save changes (Guardar cambios).

Además, si necesita añadir más usuarios a un caso de uso, consulte la sección Gestión del grupo de usuarios de Cognito.

Seleccione la configuración de red

Este paso del asistente le permite implementar el caso de uso con una Amazon Virtual Private Cloud (Amazon VPC) preexistente o nueva. Si selecciona una VPC preexistente, debe proporcionar un ID de VPC, hasta 16 ID de subred y hasta 5 grupos de seguridad IDs para usarlos con esta VPC. Si no utilizas una VPC preexistente, estos ajustes se configurarán automáticamente.

Selección de un modelo

En el paso Seleccionar modelo, puedes elegir tu proveedor de modelos en el menú desplegable. Hay dos opciones: Bedrock y. SageMaker

Si lo selecciona SageMaker, puede crear un punto final del modelo de SageMaker IA en la consola de SageMaker IA y proporcionar el esquema de entrada que el modelo espera y el resultado JSONPath para la respuesta de LLM. Puede consultar la sección Uso de Amazon SageMaker AI como proveedor de LLM y los ejemplos de carga útil de SageMaker IA que se proporcionan en el repositorio de GitHub la solución.

Si selecciona Amazon Bedrock, se le presentarán cuatro opciones:

- Modelos de inicio rápido: comience rápidamente con una colección de modelos con diferentes price/performance características. Se recomienda para crear tus primeras aplicaciones. Esta opción le permite seleccionar un nombre de modelo de la lista proporcionada.
- Otros modelos de bases: acceda a la gama completa de modelos de bases con diferentes capacidades y especializaciones. Esta opción le permite introducir el identificador del modelo de base Bedrock bajo demanda que desee.
- Perfiles de inferencia: los perfiles de inferencia aprovechan la inferencia entre regiones de Bedrock para aumentar el rendimiento y mejorar la resiliencia al enrutar las solicitudes entre varias regiones de AWS durante los picos de uso. Esta opción le permite introducir el ID del perfil de inferencia que desea utilizar.
- Modelos aprovisionados: capacidad de rendimiento dedicada para cargas de trabajo de producción que requieren un rendimiento constante. Esta opción le permite introducir el ARN del provisioned/custom modelo que se va a utilizar desde Amazon Bedrock.

El paso de selección del modelo también le permite elegir la configuración avanzada del modelo. Consulte los ajustes avanzados de LLM para obtener más información sobre la configuración de Amazon Bedrock Guardrails, el rendimiento aprovisionado para Amazon Bedrock y otros parámetros del modelo.

Inferencia entre regiones

La inferencia entre regiones ayuda a los usuarios de Amazon Bedrock a gestionar sin problemas las ráfagas de tráfico no planificadas mediante el uso de la computación en diferentes regiones de AWS. Para utilizar la inferencia entre regiones, necesita el perfil de inferencia. Un perfil de inferencia es

una abstracción de un conjunto de recursos bajo demanda de un conjunto configurado de regiones de AWS. Puede enrutar su solicitud de inferencia, que se origina en su región de origen, a otra región configurada en ese grupo. Esto permite la distribución del tráfico en varias regiones de AWS. Esto ayuda a lograr un mayor rendimiento y una mayor resiliencia durante los períodos de máxima demanda.

Los perfiles de inferencia reciben el nombre del modelo y las regiones que admiten. Debe llamar a un perfil de inferencia de una de las regiones que incluye. Por ejemplo, como se muestra en la siguiente tabla, el ID del perfil de inferencia `us.anthropic.claude-3-haiku-20240307-v1:0` permite distribuir el tráfico entre `us-east-1` las `us-west-2` regiones del modelo que elija. Algunos modelos solo están disponibles con un perfil de inferencia en una región determinada.

Perfil de inferencia	ID de perfil de inferencia	Regiones incluidas
US Anthropic Claude 3 Haiku	`us.anthropic.claude-3-haiku-20240307-v1:0`	Este de EE. UU. (Norte de Virginia) (`us-east-1`) Oeste de EE. UU. (Oregón) (`us-west-2`)

Si desea utilizar un ID de perfil de inferencia en lugar de un ID de modelo, debe identificar el ID de perfil de inferencia correspondiente. Consulte Regiones y modelos compatibles para los perfiles de inferencia en la Guía del usuario de Amazon Bedrock para obtener más información. En la consola de Amazon Bedrock, la opción de inferencia entre regiones del menú de navegación de la izquierda proporciona estos perfiles de inferencia. IDs

Tras identificar el ID del perfil de inferencia que va a utilizar, puede utilizarlo durante la etapa de selección del modelo siguiendo los siguientes pasos:

1. Seleccione Amazon Bedrock como proveedor de modelos.
2. Seleccione la opción del botón de opción Perfiles de inferencia.
3. Introduzca su ID de perfil de inferencia en el cuadro de texto que aparece.

Consulte Mejorar la resiliencia con la inferencia entre regiones en la Guía del usuario de Amazon Bedrock para obtener más información sobre los perfiles de inferencia.

Seleccione la base de conocimientos

Si quieres implementar un caso práctico de generación aumentada (RAG) sin recuperación, puedes saltarte este paso.

Sin embargo, si desea habilitar RAG como parte de su implementación, ahora puede proporcionar un ID de índice de Amazon Kendra preconfigurado o un ID de Amazon Bedrock Knowledge Base. También puede crear un nuevo índice de Amazon Kendra para usarlo con la solución. Actualmente, la solución es compatible con las bases de conocimiento de Amazon Kendra y Amazon Bedrock como bases de conocimiento para la implementación de casos de uso basados en RAG.

Consulte la sección Configuración de una base de conocimientos para obtener directrices sobre la incorporación de datos a la base de conocimientos para utilizarlos en una implementación basada en RAG.

Configuraciones RAG avanzadas

El asistente le permite seleccionar opciones avanzadas para utilizarlas con su implementación de RAG, como el número de documentos que se van a recuperar cada vez que se envía una consulta a su base de conocimientos, una respuesta de texto estático del LLM cuando no se encuentra ningún documento en la base de conocimientos, si desea mostrar las fuentes de los documentos con su respuesta de LLM para comprobar su integridad, etc. Además, también puede configurar configuraciones específicas de la base de conocimientos para Amazon Kendra, como el control de acceso basado en roles (RBAC) o la anulación del tipo de búsqueda cuando utilice Amazon OpenSearch Serverless con las bases de conocimiento de Amazon Bedrock. Consulte la sección de configuración avanzada de la base de conocimientos para obtener más información sobre estas configuraciones avanzadas.

> **Note**
>
> Su base de conocimientos debe estar en la misma cuenta y región que el panel de implementación implementado y las pilas de casos de uso.

Selecciona las indicaciones y los límites de los tokens

En este paso, puede configurar su indicador para usarlo con el LLM. Las indicaciones pueden requerir marcadores de posición como`{input}`, y. `{history}` `{context}` Estos marcadores de posición indican al LLM dónde extraer las entradas de los usuarios, el historial de conversaciones y la información extraída de la base de conocimientos.

- Para el proveedor de modelos Bedrock, se debe proporcionar el indicador del sistema, que no tiene restricciones para un caso de uso ajeno a RAG. Sin embargo, el mensaje de desambiguación para el proveedor de modelos Bedrock requiere un mínimo de dos marcadores de posición, y `{input}` `{history}`
- Para el proveedor SageMaker del modelo, el sistema y las indicaciones de desambiguación, ambos requieren un mínimo de dos marcadores de posición: y. `{input}` `{history}`
- Para los casos de uso de RAG, se requiere además el marcador de posición para cada proveedor de modelos. `{context}`

Para obtener más información, consulte Configuración de las indicaciones. También puedes consultar la sección Consejos para gestionar los límites de los tokens de los modelos y seleccionar los tamaños de los límites de los tokens para tus indicaciones.

Habilita la entrada multimodal

Este paso le permite habilitar las capacidades de entrada multimodal para su caso de uso. Cuando está habilitada, los usuarios pueden cargar y enviar imágenes y documentos junto con sus consultas de texto.

Tipos de archivos y restricciones compatibles:

- Imágenes: hasta 20 imágenes por mensaje. Cada imagen no debe tener más de 3,75 MB de tamaño y 8000 px de alto y ancho. Formatos compatibles: png, jpeg, gif, webp
- Documentos: hasta 5 documentos por mensaje. Cada documento no debe tener un tamaño superior a 4,5 MB. Formatos compatibles: pdf, csv, doc, docx, xls, xlsx, html, txt, md

Cómo utilizar la entrada multimodal:

1. Habilite el MultimodalEnabledparámetro durante la implementación del caso de uso
2. En la interfaz de chat, los usuarios pueden cargar archivos de dos maneras:
   - Al hacer clic en el botón de carga del cuadro de entrada del chat, o
   - Arrastrar y soltar archivos directamente en la interfaz de chat
3. Los archivos se cargan en Amazon S3 y el modelo seleccionado los procesa
4. Los archivos cargados se eliminan automáticamente después de 48 horas

Seguimiento del estado de los archivos:

DevOps los usuarios pueden supervisar los metadatos de los archivos en DynamoDB, que incluyen el tiempo de carga y el estado del procesamiento. Los archivos pueden tener los siguientes estados:

- pendiente: se ha iniciado la carga del archivo, pero aún no se ha completado. Este es el estado inicial cuando se genera una URL prefirmada.
- cargado: el archivo se ha cargado correctamente en S3 y está listo para que el modelo lo procese.
- eliminado: el usuario ha eliminado el archivo y ya no debería estar accesible para su procesamiento.
- no válido: el archivo no pasó las comprobaciones de validación (por ejemplo, no coincide el tipo de archivo o falla en la validación de seguridad).

Los archivos en estado pendiente que nunca se carguen se limpiarán automáticamente cuando su TTL caduque. El modelo solo puede procesar los archivos con el estado de carga.

El depósito multimodal de S3 y la tabla de metadatos de DynamoDB están disponibles en las salidas del panel de despliegue con las `MultimodalDataBucketName` claves y, respectivamente. `MultimodalDataMetadataTable`

> **Note**
>
> No todos los modelos admiten la entrada multimodal. Asegúrese de que el modelo seleccionado sea compatible con el procesamiento de imágenes y documentos antes de activar esta función. Consulte los modelos de base compatibles en la documentación de Amazon Bedrock para comprobar qué modelos admiten la imagen como modalidad de entrada.

> **Important**
>
> Los archivos subidos por los usuarios se almacenan en Amazon S3 con una política de ciclo de vida de 48 horas. Los metadatos sobre los archivos cargados se almacenan en Amazon DynamoDB con un TTL de 24 horas para el historial de conversaciones.

Revise e implemente

Tras este paso, revise la configuración que ha seleccionado y elija Deploy Use Case. A continuación, el nuevo caso de uso se despliega y se hace visible en la vista del panel de implementación para seguir gestionándolo.

Paso 3b: Implementar un caso de uso de Bedrock Agent

El caso de uso de Bedrock Agent proporciona un mecanismo potente y seguro para invocar a los agentes de Amazon Bedrock en sus casos de uso. Esta función permite a los desarrolladores integrar sin problemas las capacidades de los agentes autónomos impulsados por la IA, que pueden organizar y ejecutar tareas de varios pasos en varios modelos básicos, fuentes de datos, aplicaciones de software y conversaciones con los usuarios, al tiempo que mantienen sólidas medidas de seguridad.

Requisitos previos

Antes de crear un agente de Amazon Bedrock, asegúrate de tener lo siguiente:

1. La cuenta de AWS en la que se implementa Generative AI Application Builder en AWS, con acceso a la consola Amazon Bedrock.
2. Permisos de IAM adecuados para crear y gestionar Amazon Bedrock Agents.

Creación de un agente de Amazon Bedrock

Consulte la sección Crear y configurar un agente manualmente en la Guía del usuario de Amazon Bedrock para obtener instrucciones detalladas sobre la creación de un agente. Puede configurar opciones como:

- Instrucciones (indicaciones) para su agente
- Base de conocimientos, que se utiliza para buscar información adicional en función de las entradas del usuario
- Memoria del agente para que los agentes puedan recordar la información de varias sesiones (durante un máximo de 30 días)

Una vez que haya creado correctamente un agente de Amazon Bedrock, puede continuar con el flujo del asistente de casos de uso de Generative AI Application Builder en AWS Bedrock Agent. Para ello, elija Implementar un nuevo caso de uso en el panel de implementación y seleccione Crear un

caso de uso de Bedrock Agent. Siga las instrucciones del asistente y utilice los siguientes pasos para configurar el caso de uso.

Seleccione el caso de uso

Este paso es el mismo que el caso de uso del texto descrito anteriormente.

Seleccione la configuración de red

Este paso es el mismo que el caso de uso de Text descrito anteriormente

Seleccione un agente

En este paso, debe proporcionar el ID de agente y el ID de alias del agente de Amazon Bedrock que creó.

Paso 3c: Implementar un caso de uso de un servidor MCP

El caso de uso del servidor MCP (Model Context Protocol) le permite implementar y administrar servidores MCP que se pueden integrar con modelos y agentes de IA. Los servidores MCP proporcionan una forma estandarizada de exponer las herramientas, los recursos y las capacidades a las aplicaciones de IA. Puede crear servidores MCP a partir de funciones APIs Lambda existentes o alojar servidores MCP personalizados mediante imágenes de contenedor.

Requisitos previos

Antes de implementar un caso de uso de un servidor MCP, asegúrese de tener lo siguiente:

1. La cuenta de AWS en la que se implementa Generative AI Application Builder en AWS.
2. Permisos de IAM adecuados para crear y gestionar los recursos de Amazon Bedrock AgentCore .
3. Según el método de creación que elija:
   - Para el método Gateway (Lambda/API/MCPservidor): funciones de Lambda, puntos de enlace de API con sus archivos de esquema correspondientes (formato JSON para Lambda, OpenAPI/Smithy para APIs) o puntos de enlace URL de servidor MCP
   - Para el método Runtime (ECR): una imagen de contenedor Docker enviada a Amazon ECR que contiene la implementación de su servidor MCP

Métodos de creación del servidor MCP

La solución admite dos métodos para crear servidores MCP:

Crear desde un servidor Lambda, API o MCP (método Gateway)

Este método crea una puerta de enlace MCP que agrupa las funciones Lambda existentes, REST o servidores MCP externos APIs, lo que los hace accesibles como herramientas MCP. La puerta de enlace gestiona la traducción de protocolos entre MCP y sus servicios existentes.

- Objetivos de Lambda: integre las funciones de Lambda existentes proporcionando el ARN de la función y un archivo de esquema JSON que describa el formato de la función input/output
- Objetivos de OpenAPI: integre REST utilizando las especificaciones de APIs OpenAPI (formato JSON o YAML) con soporte para OAuth la autenticación 2.0 o API Key
- Objetivos de Smithy: integre lo APIs definido mediante archivos de modelo de Smithy (formato.smithy o.json)
- Destinos del servidor MCP: Conéctese directamente a servidores MCP externos a través de puntos finales URL, lo que permite la integración de los servidores MCP existentes sin implementar una nueva infraestructura

Puede configurar varios destinos (hasta 10) dentro de una única puerta de enlace MCP, cada uno de los cuales representa una herramienta o capacidad diferente.

Alojamiento desde una imagen ECR (método de ejecución)

Este método implementa un servidor MCP en contenedores a partir de una imagen de Amazon ECR. Utilice este enfoque cuando tenga una implementación de servidor MCP personalizada que deba ejecutarse como un servicio independiente.

- Proporcione el URI de la imagen ECR (debe incluir una etiqueta, p. ej., o) `:latest :v1.0.0`
- Si lo desea, configure las variables de entorno para transferir la configuración a su contenedor
- El contenedor debe implementar el protocolo MCP y exponer los puntos finales necesarios

Implementación de un servidor MCP

Para implementar un caso de uso de un servidor MCP, elija Implementar un nuevo caso de uso en el panel de implementación y seleccione Crear un caso de uso de servidor MCP. Siga las instrucciones del asistente y utilice los siguientes pasos para configurar el caso de uso.

Seleccione el caso de uso

Este paso es el mismo que el caso de uso del texto descrito anteriormente.

Seleccione la configuración de red

Actualmente, solo está habilitado el acceso público y la VPC no es compatible con la configuración de red.

Cree un servidor MCP

En este paso, configura la implementación del servidor MCP:

Método de creación del servidor MCP

Elija entre los dos métodos de creación:

- Crear desde un servidor Lambda, API o MCP: cree una puerta de enlace MCP a partir de funciones Lambda existentes, especificaciones de API o puntos finales de servidores MCP externos
- Alojamiento desde una imagen ECR: Implemente un servidor MCP personalizado a partir de una imagen de contenedor

> **Note**
>
> El método de creación no se puede cambiar después de la implementación. Si necesita cambiar de método, debe implementar un nuevo caso de uso del servidor MCP.

Configuración de puerta de enlace (para el método Lambda/API/MCP de servidor)

Si seleccionó el método Gateway, configure uno o más destinos:

1. Nombre de destino (obligatorio): un nombre descriptivo para identificar esta configuración de destino
2. Descripción del objetivo (opcional): una breve descripción de lo que hace este objetivo
3. Tipo de objetivo: seleccione el tipo de objetivo que desee configurar:
   - Lambda: para funciones de AWS Lambda
   - OpenAPI: para REST con especificaciones de APIs OpenAPI
   - Smithy: Para APIs las definiciones del modelo Smithy
   - Servidor MCP: para la conexión directa a servidores MCP externos a través de puntos finales de URL

4. Archivo de esquema (obligatorio): cargue el archivo de esquema que describe su objetivo:
   - Para Lambda: archivo de esquema JSON que describe input/output el formato. Para obtener más información sobre la creación de esquemas de herramientas Lambda, consulte el esquema de herramientas Lambda en la Guía para desarrolladores de Amazon Bedrock. AgentCore
   - Para OpenAPI: archivo de especificaciones de OpenAPI (JSON o YAML). Para obtener más información sobre los requisitos del esquema de OpenAPI, consulte el esquema de OpenAPI en la Guía para desarrolladores de Amazon Bedrock. AgentCore
   - Para Smithy: archivo de modelo de Smithy (.smithy o.json). Para obtener más información sobre la creación de objetivos de Smithy, consulte Creación de objetivos de Smithy en la Guía para desarrolladores de Amazon Bedrock AgentCore .
5. ARN de la función Lambda (necesaria para los objetivos de Lambda): el ARN de la función Lambda que se va a integrar
6. URL del servidor MCP (necesaria para los destinos del servidor MCP): el extremo URL del servidor MCP externo al que se va a conectar. La URL debe estar codificada correctamente y el servidor MCP debe admitir las capacidades de las herramientas con las versiones del protocolo MCP 2025-06-18. Para obtener más información, consulte los destinos de los servidores MCP en la Guía para AgentCore desarrolladores de Amazon Bedrock.
7. Autenticación saliente (necesaria para los destinos de OpenAPI): configure la autenticación para las llamadas a la API REST:
   - Tipo de autenticación: elija la clave OAuth 2.0 o la clave de API
   - ARN del proveedor de autenticación saliente: el ARN del proveedor de credenciales en la bóveda de tokens de Amazon Bedrock AgentCore
   - Configuraciones adicionales: según el tipo de autenticación:
     - Para OAuth 2.0: configure los ámbitos y los parámetros personalizados
     - Para la clave de API: especifique la ubicación (encabezado o parámetro de consulta), el nombre del parámetro y el prefijo opcional

Puede añadir varios objetivos (hasta 10) seleccionando Añadir otro objetivo. Cada objetivo representa una herramienta o capacidad independiente expuesta por su servidor MCP.

Configuración ECR (para el método de imagen ECR)

Si seleccionó el método Runtime, proporcione:

1. URI de imagen ECR (obligatorio): el URI completo de la imagen de Docker en Amazon ECR

- Formato: `account-id.dkr.ecr.region.amazonaws.com/repository-name:tag`
- La imagen debe estar en la misma región de AWS que su implementación
- Se requiere una etiqueta (por ejemplo`:latest`,`:v1.0.0`)

2. Variables de entorno (opcional): configura pares clave-valor para pasarlos a tu contenedor en tiempo de ejecución
   - Utilícelas para proporcionar configuraciones, credenciales o indicadores personalizados
   - Puede añadir hasta 10 variables de entorno

Revise e implemente

Tras configurar el servidor MCP, revise la configuración que ha seleccionado y elija Deploy Use Case. A continuación, el nuevo caso de uso del servidor MCP se despliega y pasa a ser visible en la vista del panel de implementación para su posterior administración.

> **Note**
>
> Las implementaciones de MCP Server crean recursos en Amazon Bedrock AgentCore, incluidas las puertas de enlace, los tiempos de ejecución y las identidades de las cargas de trabajo. La solución administra estos recursos automáticamente y se eliminarán al eliminar el caso de uso.

Paso 3d: implementar un caso de uso de Agent Builder

El Agent Builder le permite crear, configurar e implementar agentes de IA listos para la producción en Amazon Bedrock. AgentCore Esta función proporciona un control total sobre el comportamiento de los agentes mediante las indicaciones del sistema, la selección del modelo, la integración del servidor MCP y la administración de la memoria.

El proceso de implementación es básicamente el mismo que en un caso de uso de Text, con algunas diferencias notables.

Seleccione un caso de uso

Este paso es el mismo que el caso de uso del texto descrito anteriormente.

Detalles del caso de uso

Este paso es el mismo que el caso de uso del texto descrito anteriormente.

Configurar el agente

En este paso, configurará los ajustes principales del agente, incluida la línea de comandos del sistema, servers/Strands las herramientas MCP disponibles y la memoria.

Símbolo del sistema

El mensaje del sistema define el comportamiento, la personalidad y las capacidades del agente. Puede hacer lo siguiente:

- Editar la plantilla de solicitud del sistema predeterminada
- Utilice el botón Restablecer valores predeterminados para restaurar la plantilla original
- Incluya instrucciones para el uso de la herramienta y el formato de las respuestas

Integración del servidor MCP (opcional)

Configure los servidores del Model Context Protocol para proporcionar a su agente acceso a las herramientas y los datos empresariales:

1. Seleccione uno de los servidores MCP disponibles en el menú desplegable
2. Revise las herramientas listas para usar a las que podrá acceder el agente

> **Note**
>
> Los servidores MCP deben estar configurados y ser accesibles antes de la implementación. Consulte la documentación del MCP para ver las instrucciones de configuración del servidor.

Configuración de memoria

Configure la forma en que el agente mantiene el contexto y el conocimiento:

- Memoria a corto plazo: habilitada de forma predeterminada para todos los agentes. Mantiene el contexto de la conversación dentro de las sesiones.
- Memoria a largo plazo: active esta opción para permitir la extracción y el almacenamiento de información en todas las sesiones. Utiliza AgentCore la memoria con una estrategia de memoria semántica.

Revisa e implementa

Tras este paso, revise la configuración que ha seleccionado y elija Deploy Use Case. La implementación de Agent Builder normalmente se completa en 10 a 15 minutos. A continuación, el nuevo caso de uso pasa a estar visible en la vista del panel de implementación para seguir gestionándolo.

Paso 3e: Implementar un caso de uso de Workflow

El generador de flujos de trabajo le permite crear agentes supervisores que orquesten varios agentes de Agent Builder utilizando el patrón de delegación de agentes como herramientas. Esta función le permite crear flujos de trabajo complejos con varios agentes mediante la reutilización de las implementaciones de Agent Builder existentes.

El proceso de despliegue sigue un patrón similar al de Agent Builder, con pasos adicionales para la detección y selección de los agentes.

Seleccione un caso de uso

Este paso es el mismo que el caso de uso del texto descrito anteriormente.

Detalles del caso de uso

Este paso es el mismo que el caso de uso del texto descrito anteriormente.

Configure el agente supervisor

En este paso, debe configurar el agente supervisor que coordinará a los agentes especializados de Agent Builder.

Símbolo del sistema

El indicador del sistema define la forma en que el agente supervisor delega el trabajo a los agentes especializados. Puede:

- Editar la plantilla de solicitud del sistema predeterminada
- Incluya instrucciones para la selección y delegación de agentes
- Defina cómo agregar los resultados de varios agentes
- Utilice el botón Restablecer valores predeterminados para restaurar la plantilla original

> **Note**
>
> El indicador del sistema debe describir claramente cuándo y cómo utilizar cada agente especializado. Las descripciones de los agentes son fundamentales para una delegación adecuada.

Selección de modelos

Seleccione el modelo base para el agente supervisor. El agente supervisor usa este modelo para:

- Comprenda las solicitudes de los usuarios
- Seleccione los agentes especializados adecuados
- Coordine la ejecución del agente
- Agregue y formatee las respuestas

Seleccione agentes especializados

En este paso, selecciona a qué agentes de Agent Builder puede delegar el trabajo el supervisor.

Añadir agentes

1. Haga clic en Añadir agente para abrir el cuadro de diálogo de selección de agentes
2. Seleccione uno o más agentes de Agent Builder de la lista
3. Revise las descripciones de los agentes que se le proporcionarán al supervisor
4. Confirme la selección

> **Note**
>
> - Los flujos de trabajo requieren al menos un caso de uso de Agent Builder como agente especializado
> - Todos los agentes especializados deben desplegarse correctamente antes de crear el flujo de trabajo

Revise e implemente

Revise la configuración del flujo de trabajo, que incluye:

- Modelo y indicador del sistema Supervisor Agent
- Lista de agentes especializados
- Configuración de memoria

Elija Deploy Use Case. La implementación del flujo de trabajo normalmente se completa en 15 a 20 minutos. El nuevo flujo de trabajo aparece en la vista del panel de implementación para seguir gestionándolo.

Paso 4: Configuración posterior a la implementación

En esta sección se proporcionan recomendaciones para configurar la solución después de la implementación.

Control de versiones de buckets de Amazon S3, políticas de ciclo de vida y replicación entre regiones

Esta solución no impone las configuraciones del ciclo de vida de los buckets que crea. Le recomendamos lo siguiente:

- Establecer configuraciones de ciclo de vida para las implementaciones de producción. Para obtener más información, consulte Establecer la configuración del ciclo de vida en un depósito en la Guía del usuario de Amazon Simple Storage Service.
- Habilitar el control de versiones y la replicación entre regiones para los buckets de Amazon S3 en función del caso de uso para el que se implemente la solución.

Copias de seguridad de Amazon DynamoDB

Esta solución utiliza DynamoDB para varios fines (consulte los servicios de AWS en esta solución). La solución no habilita las copias de seguridad de las tablas que crea. Recomendamos crear una copia de seguridad de esta función para las implementaciones de producción. Consulte Hacer copias de seguridad de una tabla de DynamoDB y Utilizar AWS Backup for DynamoDB para obtener más información.

CloudWatch Panel de control y alarmas de Amazon

La solución implementa un panel personalizado CloudWatch para representar gráficos a partir de métricas publicadas personalizadas y métricas de servicios de AWS. Recomendamos crear CloudWatch alarmas y añadir notificaciones en función del caso de uso para el que se implemente la solución.

Amazon CloudWatch Logs

Los registros de Lambda se configuran para que no caduquen nunca y los registros de API Gateway se configuran con una caducidad de 10 años. Puede actualizar la caducidad de los grupos de registros respectivos para adaptarlos a la política de retención de registros de su empresa.

Dominios web personalizados con certificados TLS v1.2 o superior

La solución implementa una interfaz de usuario web y una API Gateway optimizada para Edge mediante CloudFront. CloudFrontSu dominio no exige certificados TLS v1.2 o superiores. Recomendamos crear un dominio personalizado con Amazon Route 53, crear un certificado con AWS Certificate Manager o usar un certificado existente si su organización tiene uno.

Para obtener más información, consulte la Guía para desarrolladores de Amazon Route 53 y Cómo elegir una versión mínima de TLS para un dominio personalizado en API Gateway.

Escalar con Amazon Kendra

Esta solución ofrece la posibilidad de utilizar Amazon Kendra para realizar búsquedas inteligentes basadas en la PNL en los documentos ingeridos. Puede aumentar la capacidad de Amazon Kendra mediante los siguientes CloudFormation parámetros para cargas de trabajo más grandes:

Parámetro	Predeterminado	Description (Descripción)
Capacidad de consulta adicional de Amazon Kendra	`0`	La cantidad de capacidad de consulta adicional para un índice y una GetQuerySuggestionscapacidad. Una unidad de capacidad adicional para un índice proporciona aproximadamente 8000 consultas por día.

Parámetro	Predeterminado	Description (Descripción)
Capacidad de almacenamiento adicional de Amazon Kendra	`0`	La cantidad de capacidad de almacenamiento adicional para un índice. Una sola unidad de capacidad proporciona 30 GB de espacio de almacenamiento o 100 000 documentos, lo que ocurra primero.
Edición Amazon Kendra	`Developer`	Amazon Kendra ofrece las ediciones Developer y Enterprise para crear índices. Para obtener más información sobre las diferencias entre las ediciones Amazon Kendra, consulte los precios de Amazon Kendra.

Para modificar los valores de estos CloudFormation parámetros, seleccione los valores adecuados en el momento de implementar la pila. Para obtener más información sobre las unidades de capacidad de consulta y almacenamiento, consulte Ajustar la capacidad.

> **Note**
>
> Si el caso de uso de Text no se implementa con RAG activado, no se utiliza ni se crea un índice de Amazon Kendra.

Configuración del SSO mediante la federación de Idp

Esta solución permite la integración con proveedores de identidad externos que admiten la federación de identidades basada en SAML u OIDC. Cuando la solución se implementa, crea un grupo de usuarios de Amazon Cognito y una integración de clientes de aplicaciones individuales para el panel de implementación y los casos de uso individuales. En función del IDP externo, siga los pasos que se indican en la sección Configuración de proveedores de identidad para su grupo de

usuarios de la Guía para desarrolladores de Amazon Cognito y elija la integración entre el cliente y la aplicación para el panel de implementación o el caso de uso con el que desee configurar el SSO.

Para pasar la información del grupo de usuarios a la base de conocimientos o a los almacenes de vectores en una arquitectura basada en RAG, necesitará mapear los grupos de usuarios del IDP externo a los grupos de usuarios de Amazon Cognito. La solución proporciona un activador inicial de la función Lambda de andamiaje que se mapea con la fase previa a la generación del token. La función Lambda tiene el archivo group_mapping.json que debe actualizarse para proporcionar las asignaciones de grupos. Consulte Personalización de los flujos de trabajo de grupos de usuarios con activadores de Lambda para ver los activadores de Lambda compatibles con Amazon Cognito.

Configuración manual del grupo de usuarios

Si decide no pasar el correo electrónico de un administrador o de un usuario predeterminado durante la implementación, debe crear manualmente los grupos de usuarios correspondientes en Amazon Cognito para garantizar los permisos correctos:

1. Para el panel de implementación, cree un grupo con un nombre `Admin` en su grupo de usuarios de Cognito.
2. Para cada caso de uso, cree un grupo con un nombre `${UseCaseName}-Users` en su grupo de usuarios de Cognito, donde `${UseCaseName}` aparece el nombre del caso de uso implementado.

Estos grupos son necesarios para que el mecanismo de autorización funcione correctamente. Todos los usuarios a los que desee conceder acceso deben agregarse a los grupos correspondientes.

Si `placeholder@example.com` se aprueba, se creará el grupo de Cognito, pero deberá seguir creando los usuarios asociados y asignarlos al grupo.

Personalización de la pantalla de inicio de sesión

Esta solución utiliza la interfaz de usuario alojada en Amazon Cognito para representar la página de inicio de sesión. Para personalizar la página de inicio de sesión integrada, consulte Personalización de las páginas web de inicio de sesión y registro integradas en la Guía para desarrolladores de Amazon Cognito.

Consideraciones adicionales de seguridad

Según el caso de uso para el que implemente la solución, revise las siguientes recomendaciones de seguridad:

- Claves de cifrado de AWS KMS administradas por el cliente: la solución utiliza las claves de AWS KMS administradas por AWS de forma predeterminada, ya que están disponibles sin coste adicional. Revise su caso de uso para determinar si debe actualizar la solución para usar claves de AWS KMS administradas por el cliente.
- Reglas de regulación de API Gateway: la solución se implementa con las reglas de regulación predeterminadas en API Gateway. En función de su caso de uso y de los volúmenes de transacciones esperados, le recomendamos que configure la limitación para. APIs Para obtener más información, consulte las solicitudes de API Throttle para mejorar el rendimiento en la Guía para desarrolladores de Amazon API Gateway.
- Habilitar AWS CloudTrail: como práctica de seguridad recomendada, considere habilitar AWS CloudTrail en la cuenta de AWS en la que se implementa la solución para registrar las llamadas a la API en la cuenta de AWS. Para obtener más información, consulte la Guía del CloudTrail usuario de AWS.
- Detección de desviaciones: recomendamos configurar la detección de desviaciones en las CloudFormation pilas para identificar los cambios involuntarios o malintencionados en la pila de soluciones implementada y recibir notificaciones al respecto. Para obtener más información, consulte Implementación de una alarma para detectar automáticamente la desviación en las CloudFormation pilas de AWS.
- Cognito JSON Web Tokens (JWTs): la solución utiliza los tokens de Amazon Cognito para autenticarse con los puntos de enlace de la API JWTs REST. Configuramos la solución con un vencimiento de cinco minutos para los tokens de identificación y los tokens de acceso. Cuando un usuario cierra sesión, se revoca su capacidad de generar nuevos tokens (se revoca el token de actualización). Sin embargo, hasta que caduque el token actual, todas las solicitudes al punto final de la API se autenticarán correctamente, ya que tienen un token válido. Revisa las consideraciones de seguridad de tu caso de uso y ajusta el período de validez del token.

Personalización de las políticas del ciclo de vida:

Para las implementaciones de producción, revise y ajuste las políticas del ciclo de vida en función de sus requisitos de retención. Consulte Establecer la configuración del ciclo de vida en un depósito en la Guía del usuario de Amazon Simple Storage Service.

Ciclo de vida y almacenamiento de archivos multimodales

Si ha activado las capacidades de entrada multimodal (MultimodalEnabledconfiguradas en`Yes`) para su caso de uso, la solución crea un bucket de Amazon S3 para almacenar los archivos cargados y una tabla de DynamoDB para realizar un seguimiento de los metadatos de los archivos.

Políticas de ciclo de vida predeterminadas:

- Archivos S3: se eliminan automáticamente después de 48 horas
- Metadatos de DynamoDB: los registros caducan después de 24 horas (historial de conversaciones TTL)

Consideraciones de seguridad:

- Los archivos se dividen por ID de caso de uso, ID de usuario, ID de conversación e ID de mensaje y, en su lugar, un archivo se almacena con un nombre UUID. La asignación del UUID a los nombres de los archivos está disponible en la tabla de metadatos de DynamoDB.
- Los usuarios solo pueden acceder a los archivos que hayan subido en sus propias conversaciones
- La validación del tipo de archivo se realiza mediante la detección de números mágicos
- Recomendamos activar Amazon GuardDuty Malware Protection for S3 para analizar los archivos cargados en busca de contenido malicioso.

Implementación de un caso de uso de Text independiente

Siga las step-by-step instrucciones de esta sección para configurar e implementar la solución en su cuenta.

Tiempo de implementación: aproximadamente de 10 a 30 minutos

1. Inicie sesión en la consola de administración de AWS y seleccione el botón para lanzar la CloudFront plantilla que desea implementar.

BedrockChat.plantilla	Launch solution
SageMakerChat.plantilla	Launch solution

2. La plantilla se lanza en la región Este de EE. UU. (Norte de Virginia) de forma predeterminada. Para lanzar la solución en una región de AWS diferente, utilice el selector de regiones de la barra de navegación de la consola.

   Nota: Esta solución utiliza Amazon Kendra y Amazon Bedrock, que actualmente no están disponibles en todas las regiones de AWS. Si utiliza estas funciones, debe lanzar esta solución en una región de AWS en la que estén disponibles estos servicios. Para obtener la disponibilidad más reciente por región, consulte la lista de servicios regionales de AWS.

3. En la página Crear pila *, compruebe que la URL de la plantilla correcta esté en el cuadro de texto *URL de Amazon S3 *y seleccione *Siguiente.
4. En la página *Especifique los detalles de la pila *, asigne un nombre a la pila de soluciones. Para obtener información sobre las limitaciones de nombres de caracteres, consulte los límites de IAM y STS en la Guía del usuario de AWS Identity and Access Management.
5. En Parámetros, revise los parámetros de esta plantilla de solución y modifíquelos según sea necesario. Esta solución utiliza los siguientes valores predeterminados.

UseCaseUUID (Identificador único universal)	*<_Requires input_>*	36 caracteres UUIDv4 para identificar este caso de uso implementado en una aplicación.
UseCaseConfigRecordKey	*<_Requires input_>*	Clave correspondiente al registro que contiene las configuraciones requeridas por el proveedor de chat Lambda en tiempo de ejecución. El registro de la tabla debe tener un atributo clave que coincida con

		este valor y un atributo de configuración que contenga la configuración deseada. La plataforma de despliegue rellenará este registro si está en uso. Para las implementaciones independientes de este caso de uso, se requiere una entrada creada manualmente en la tabla definida en UseCaseConfigTableName.
UseCaseConfigTableName	*<_Requires input_>*	La pila leerá la configuración de la tabla con este nombre en la clave UseCaseConfigRecordKey

ExistingRestApiId	(Entrada opcional)	ID de API REST de API Gateway existente que se debe utilizar. Si no se proporciona, se creará una nueva API REST de API Gateway. Por lo general, se proporciona cuando se implementa desde el panel de implementación. Nota: El uso de APIs Existing puede ayudar a reducir la duplicación de recursos y a simplificar la administración APIs cuando se necesitan implementar varios casos de uso independientes. Al suministrar los existentes APIs para un caso de uso independiente, usted es responsable de asegurarse de que la API esté configura da con las rutas requeridas con los modelos esperados. Es necesario configurar una ruta /details preconfigurada (que recopila los detalles del caso de uso durante el chat) y, opcionalmente, una ruta /feedback (si FeedbackE nabledestá configurada para `Yes` permitir la recopilac ión de comentarios para las respuestas del chat de LLM). Además, y también se ExistingApiRootRes

		ourceIddebe proporcionar ExistingCognitoUserPoolId. ExistingCognitoGroupPolicyTableName
ExistingApiRootResourceId	(Entrada opcional)	ID de recurso raíz de la API REST de API Gateway existente que se va a utilizar. El ID de recurso raíz de la API REST se puede obtener en la consola de AWS seleccionando el recurso raíz (/) en la sección «Recursos » de la API. A continuación, el ID del recurso se mostrará en el panel de detalles del recurso. También puedes ejecutar una llamada a la API de descripción en tu API REST para buscar el ID del recurso raíz.
FeedbackEnabled	No	Si se establece en No, la pila de casos de uso implement ada no tendrá acceso a la función de comentarios.
ExistingModelInfoTableName	(Entrada opcional)	Nombre de tabla de DynamoDB para la tabla que contiene la informaci ón del modelo y los valores predeterminados. Lo utiliza la plataforma de despliegu e. Si se omite, se creará una nueva tabla para alojar los valores predeterminados del modelo.

DefaultUserEmail	`placeholder@example.com`	Correo electrónico del usuario predeterminado para este caso de uso. Se crea un usuario de Amazon Cognito para este correo electrónico para acceder al caso de uso. Si no se proporciona, no se crearán el grupo ni el usuario de Cognito. También puede utilizarlos `placeholder@example.com` para crear el grupo, pero no el usuario. Consulte la configuración manual del grupo de usuarios para obtener información sobre cómo configurar el grupo de usuarios.
ExistingCognitoUserPoolId	(Entrada opcional)	UserPoolIdde un grupo de usuarios de Amazon Cognito existente con el que se autenticará este caso de uso. Por lo general, se proporciona cuando se implementa desde el panel de implementación, pero se puede omitir al implementar esta pila de casos de uso de forma independiente.

CognitoDomainPrefix	(Entrada opcional)	Introduzca un valor si desea proporcionar un dominio para el cliente del grupo de usuarios de Cognito. Si no proporciona un valor, la implementación generará uno.
ExistingCognitoUserPoolClient	(Entrada opcional)	Proporcione un cliente de grupo de usuarios (App Client) para usar uno existente. Si no proporciona un cliente de grupo de usuarios, se creará uno nuevo. Este parámetro solo se puede proporcionar si se proporciona un ID de grupo de usuarios existente.
ExistingCognitoGroupPolicyTableName	(Entrada opcional)	Nombre de la tabla de DynamoDB que contiene las políticas del grupo de usuarios. Lo utiliza el autorizador personalizado de la API del caso de uso. Por lo general, puede proporcionar una entrada al implementar desde la plataforma de implementación, pero puede omitirla cuando implementa esta pila de casos de uso de forma independiente.

RAGEnabled	`true`	Si se establece en true, la pila de casos de uso implementada utiliza el índice de Amazon Kendra proporcionado, creado para proporcionar la funcionalidad RAG. Si se establece en`false`, el usuario interactúa directamente con el LLM.
KnowledgeBaseType	Bedrock	Tipo de base de conocimientos que se utilizará para RAG. Configúrelo solo si lo RAGEnabled es`true`. Puede ser Bedrock o Kendra. Nota: Solo es relevante si RAGEnabledes cierto.
ExistingKendraIndexId	(Entrada opcional)	ID de índice de un índice de Kendra existente que se utilizará en el caso de uso. Si no se proporciona ninguno y KnowledgeBaseTypees Kendra, se creará un nuevo índice para usted. Nota: Solo es relevante si RAGEnabledes `true` y Knowledge BaseTypees`Kendra`.

NewKendraIndexName	(Entrada opcional)	Nombre del nuevo índice de Kendra que se va a crear para este caso de uso. Solo se aplica si no ExistingKendraIndexIdse proporciona. Nota: Solo es relevante si RAGEnabledes verdadera y KnowledgeBaseTypees Kendra.
NewKendraQueryCapacityUnits	0	Unidades de capacidad de consulta adicionales para el nuevo índice de Amazon Kendra que se va a crear para este caso de uso. Solo se aplica si no ExistingKendraIndexIdse suministra, consulte CapacityUnitsConfiguration. Nota: Solo es relevante si RAGEnabledes `true` y Knowledge BaseTypeesKendra.

NewKendraStorageCapacityUnits	`0`	Unidades de capacidad de almacenamiento adicionales para el nuevo índice de Amazon Kendra que se va a crear para este caso de uso. Solo se aplica si no ExistingKendraIndexIdse suministra, consulte CapacityUnitsConfiguration. Nota: Solo es relevante si RAGEnabledes `true` y Knowledge BaseTypeesKendra.
NewKendraIndexEdition	(Entrada opcional)	La edición de Amazon Kendra que se utilizará en el nuevo índice de Amazon Kendra que se creará para este caso de uso. Solo se aplica si no ExistingKendraIndexIdse suministra, consulte Amazon Kendra Editions. Nota: Solo es relevante si RAGEnabledes `true` y Knowledge BaseTypeesKendra.

BedrockKnowledgeBaseId	(Entrada opcional)	ID de la base de conocimientos básica que se utilizará en un caso de uso de RAG. No se puede proporcionar si se proporciona ExistingKendraIndexIdo NewKendraIndexNamese proporciona. Nota: Solo es relevante si RAGEnabledes `true` y Knowledge BaseTypeesBedrock.
VpcEnabled	No	¿Deberían implementarse los recursos de la pila en una VPC?
CreateNewVpc	No	Seleccione Yes esta opción si desea que la solución cree una nueva VPC para usted y se utilice en este caso de uso. Nota: Solo es relevante si lo VpcEnabledesYes.
IPAMPoolId	(Entrada opcional)	Si desea asignar el rango CIDR mediante el administrador de direcciones IP de Amazon VPC, proporcione el ID del grupo de IPAM que va a utilizar. Nota: Solo es relevante si VpcEnabledes y esYes. CreateNewVpcNo

ExistingVpcId	(Entrada opcional)	ID de VPC de una VPC existente que se utilizará en el caso de uso. Nota: Solo es relevante si VpcEnabledes `Yes` y CreateNewVpces. `No`
ExistingPrivateSubnetIds	(Entrada opcional)	Lista separada por comas de subredes IDs de subredes privadas existentes que se utilizarán para implementar la función Lambda. Nota: Solo es relevante si VpcEnabledes y es. `Yes` CreateNewVpc`No`
ExistingSecurityGroupIds	(Entrada opcional)	Lista separada por comas de los grupos de seguridad de la VPC existente que se utilizarán para configurar las funciones de Lambda. Nota: Solo es relevante si VpcEnabledes y es`Yes`. CreateNewVpc`No`
VpcAzs	(Entrada opcional)	Lista separada por comas de AZs en la que se crean las subredes del VPCs Nota: Solo es relevante si VpcEnabledes `Yes` y CreateNewVpces. `No`

UseInferenceProfile	No	Si el modelo configurado es Bedrock, puede indicar si está utilizando el perfil de inferencia de Bedrock. Esto garantizará que las políticas de IAM requerida s se configuren durante el despliegue del stack. Para obtener más información, consulte el siguiente archivo - region-inference.html https:// docs.aws.amazon.com/ bedrock/ latest/userguide/c ross
Implemente la interfaz	Sí	Seleccione la opción de implementar la interfaz de usuario de la interfaz de usuario para esta implement ación. Si selecciona No, solo se creará la infraestructura que alojará el procesamiento APIs, la autenticación y el APIs backend.

6. Elija Siguiente.
7. En la página Configurar opciones de pila, elija Siguiente.
8. En la página Revisar, revise y confirme la configuración. Seleccione la casilla para confirmar que la plantilla creará los recursos de AWS Identity and Access Management (IAM).
9. Elija Create stack (Crear pila) para implementar la pila.

Puede ver el estado de la pila en la CloudFormation consola de AWS en la columna Estado. Debería recibir el estado CREATE_COMPLETE en un plazo aproximado de 10 a 30 minutos.

Implementación de un caso de uso de Bedrock Agent independiente

Siga las step-by-step instrucciones de esta sección para configurar e implementar la solución en su cuenta.

Tiempo de implementación: aproximadamente de 10 a 30 minutos

1. Inicie sesión en la consola de administración de AWS y seleccione el botón para lanzar la CloudFront plantilla.

BedrockAgent.plantilla	Launch solution

2. La plantilla se lanza en la región Este de EE. UU. (Norte de Virginia) de forma predeterminada. Para lanzar la solución en una región de AWS diferente, utilice el selector de regiones de la barra de navegación de la consola.

> **Note**
>
> Esta solución utiliza Amazon Bedrock, que actualmente no está disponible en todas las regiones de AWS. Si utiliza estas funciones, debe lanzar esta solución en una región de AWS en la que estén disponibles estos servicios. Para obtener la disponibilidad más reciente por región, consulte la lista de servicios regionales de AWS.

3. En la página Crear pila, verifique que la dirección URL de la plantilla correcta se encuentre en el cuadro de texto URL de Amazon S3 y elija Siguiente.
4. En la página Especificar los detalles de la pila, especifique un nombre para la pila. Para obtener información sobre las limitaciones de caracteres en los nombres, consulte {https---docs-aws-amazon-com- https---docs-aws-amazon-com -IAM-latest- UserGuide -reference-iam-limits-html} [Cuotas de IAM y AWS STS] en la Guía del usuario de AWS Identity and Access Management.
5. En Parámetros, revise los parámetros de esta plantilla de solución y modifíquelos según sea necesario. Esta solución utiliza los siguientes valores predeterminados.

Parámetro	Entrada predeterminada	Description (Descripción)
UseCaseUUID (Identificador único universal)	*<_Requires input_>*	36 caracteres UUIDv4 para identificar este caso de uso implementado en una aplicación.
UseCaseConfigRecordKey	*<Requires input>*	Clave correspondiente al registro que contiene las configuraciones requeridas por la función Lambda del proveedor de chat en tiempo de ejecución. El registro de la tabla debe tener un atributo clave que coincida con este valor y un atributo de configuración que contenga la configuración deseada. La plataforma de despliegue rellenará este registro si está en uso. Para las implementaciones independientes de este caso de uso, se requiere una entrada creada manualmente en la tabla definida en UseCaseConfigTableName.

Parámetro	Entrada predeterminada	Description (Descripción)
UseCaseConfigTableName	*<Requires input>`*	La pila leerá la configuración de los casos de uso en la tabla que se proporciona aquí y utilizará la clave de registro definida en. UseCaseConfigRecordKey
DefaultUserEmail	`placeholder@example.com`	Correo electrónico del usuario predeterminado para este caso de uso. La solución crea un usuario de Amazon Cognito para que este correo electrónico acceda al caso de uso.

Parámetro	Entrada predeterminada	Description (Descripción)
ExistingRestApiId	(Entrada opcional)	ID de API REST de API Gateway existente que se debe utilizar. Si no se proporciona, se creará una nueva API REST de API Gateway. Por lo general, se proporciona cuando se implementa desde el panel de implementación. Nota: El uso de APIs Existing puede ayudar a reducir la duplicación de recursos y a simplificar la administración APIs cuando se necesitan implementar varios casos de uso independientes. Al suministrar los existentes APIs para un caso de uso independiente, usted es responsable de asegurarse de que la API esté configura da con las rutas requeridas con los modelos esperados. Es necesario configurar una ruta /details preconfigurada (que recopila los detalles del caso de uso durante el chat) y, opcionalmente, una ruta /feedback (si FeedbackE nabledestá configurada para `Yes` permitir la recopilac ión de comentarios para las respuestas del chat de LLM). Además, y también

Parámetro	Entrada predeterminada	Description (Descripción)
		se ExistingApiRootRes ourceIddebe proporcionar ExistingCognitoUserPoolId. ExistingCognitoGroupPolicyT ableName
ExistingApiRootResourceId	(Entrada opcional)	ID de recurso raíz de la API REST de API Gateway existente que se va a utilizar. El ID de recurso raíz de la API REST se puede obtener en la consola de AWS seleccionando el recurso raíz (/) en la sección «Recursos » de la API. El ID del recurso se mostrará entonces en el panel de detalles del recurso. También puede ejecutar una llamada a la API de descripci ón en su API de REST para buscar el ID del recurso raíz.
FeedbackEnabled	No	Si se establece en No, la pila de casos de uso implement ada no tendrá acceso a la función de comentarios.
CognitoDomainPrefix	(Entrada opcional)	Introduzca un valor si desea proporcionar un dominio para el cliente del grupo de usuarios de Amazon Cognito. Si no proporciona un valor, la solución generará uno.

Parámetro	Entrada predeterminada	Description (Descripción)
ExistingCognitoUserPoolId	(Entrada opcional)	UserPoolIdde un grupo de usuarios de Amazon Cognito existente con el que desee autenticar este caso de uso. NOTA: Por lo general, proporciona este ID al realizar la implementación desde el panel de implement ación, pero puede omitirlo al implementar esta pila de casos de uso de forma independiente.
ExistingCognitoUserPoolClie nt	(Entrada opcional)	Proporcione un cliente de grupo de usuarios (cliente de aplicaciones) para usar uno existente. Si no proporcio na un cliente de grupo de usuarios, la solución crea uno. Solo puede proporcionar este parámetro si proporcio nó un ExistingCognitoUse rPoolId.

Parámetro	Entrada predeterminada	Description (Descripción)
ExistingCognitoGroupPolicyTableName	(Entrada opcional)	Nombre de la tabla de DynamoDB que contiene las políticas del grupo de usuarios. Lo utiliza el autorizador personalizado de la API del caso de uso. NOTA: Normalmente, se proporciona este nombre cuando se implementa desde el panel de implementación, pero se puede omitir cuando se implementa esta pila de casos de uso de forma independiente.
VpcEnabled	No	Si los recursos de la pila se implementarán en una VPC.
CreateNewVpc	No	Seleccione Yes si desea que la solución cree una nueva VPC para usted y la utilice para este caso de uso. NOTA: Este parámetro solo es relevante si lo VpcEnable desYes.
IPAMPoolId	(Entrada opcional)	Si desea asignar el rango de CIDR mediante el IPAM, proporcione el ID del grupo de IPAM que se va a utilizar. NOTA: Este parámetro solo es relevante si es y VpcEnabledesYes. CreateNewVpcNo

Parámetro	Entrada predeterminada	Description (Descripción)
ExistingVpcId	(Entrada opcional)	ID de VPC de una VPC existente que se utilizará en el caso de uso. NOTA: Este parámetro solo es relevante si VpcEnabledes `Yes` y CreateNewVpces. `No`
ExistingPrivateSubnetIds	(Entrada opcional)	Lista separada por comas de subredes IDs de subredes privadas existentes que se utilizarán para implementar la función Lambda. NOTA: Este parámetro solo es relevante si VpcEnabledes y es. `Yes` CreateNewVpcNo
ExistingSecurityGroupIds	(Entrada opcional)	Lista separada por comas de los grupos de seguridad de la VPC existente que se utilizarán para configurar las funciones de Lambda. NOTA: Este parámetro solo es relevante si VpcEnabledes y es. `Yes` CreateNewVpcNo
VpcAzs	(Entrada opcional)	Lista separada por comas de AZs en la que se crean las subredes del VPCs Nota: Solo es relevante si VpcEnabledes `Yes` y CreateNewVpces. `No`
BedrockAgentId	*`<Requires input>`*	El ID del agente de Amazon Bedrock que se va a utilizar.

Parámetro	Entrada predeterminada	Description (Descripción)
BedrockAgentAliasId	*<Requires input>*	El seudónimo del agente de Amazon Bedrock que se va a utilizar.
Implemente la interfaz	`Yes`	Seleccione la opción de implementar la interfaz de usuario de chat de la interfaz para esta implementación. La selección No da como resultado la creación de la infraestructura para alojar el procesamiento APIs, la autenticación y el APIs back-end sin la interfaz de usuario del chat.

6. Elija Siguiente.
7. En la página Configurar opciones de pila, elija Siguiente.
8. En la página Revisar, revise y confirme la configuración. Seleccione la casilla para aceptar que la plantilla creará recursos de IAM.
9. Elija Create stack (Crear pila) para implementar la pila.

Puede ver el estado de la pila en la CloudFormation consola de AWS en la columna Estado. Debería recibir el estado CREATE_COMPLETE en un plazo aproximado de 10 a 30 minutos.

Suministro de una configuración de chat de DynamoDB

Al implementar un caso de uso, UseCaseConfigRecordKeyUseCaseConfigTableNameson CloudFormation parámetros obligatorios que normalmente se rellenan en el panel de implementación. La pila de paneles de despliegue se encarga de la creación y configuración de esta tabla, mientras que las llamadas a la API de despliegue activan el rellenado de los parámetros.

Al realizar una implementación independiente, debe hacer lo siguiente:

1. Cree una tabla de DynamoDB con una clave hash de clave.

2. Cree un registro en la tabla que contenga la configuración del caso de uso como registro del formato: `{key: some_use_case_key, config: {your_configuration}.`
3. Transfiera los parámetros elegidos UseCaseConfigTableNamey UseCaseConfigRecordKey(`some_use_case_key`en este ejemplo) a la pila de casos de uso al realizar la implementación.

Para crear una configuración adecuada para una implementación independiente, puede crear un caso de uso obligatorio desde el panel de implementación y copiar el registro de la tabla de configuración. De lo contrario, puede crear su propia configuración basándose en el siguiente ejemplo para una implementación de 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
 }
}
```

Supervise la solución con Service Catalog AppRegistry

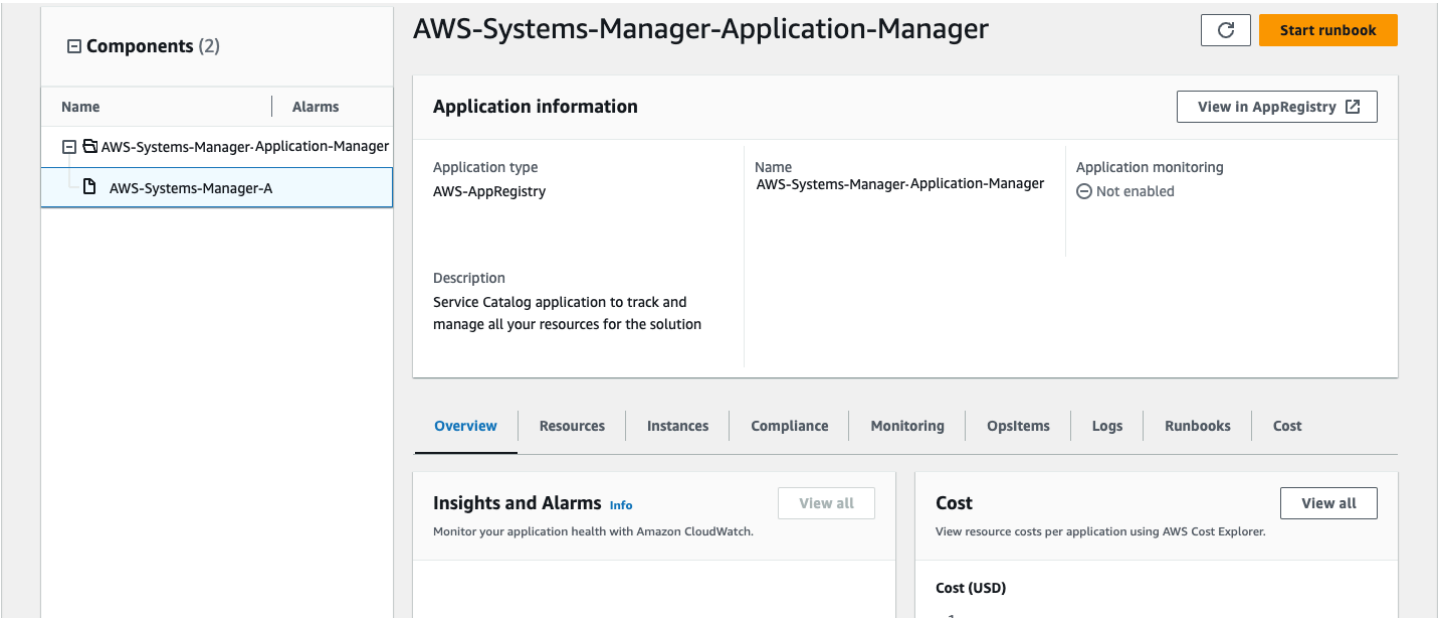
La solución incluye un AppRegistry recurso de Service Catalog para registrar la CloudFormation plantilla y los recursos subyacentes como una aplicación tanto en Service Catalog AppRegistry como en Systems Manager Application Manager.

Systems Manager Application Manager le ofrece una visión a nivel de aplicación de esta solución y sus recursos para que pueda:

- Supervise sus recursos, los costes de los recursos implementados en todas las pilas y cuentas de AWS y los registros asociados a esta solución desde una ubicación central.
- Vea los datos de operaciones de los recursos de esta solución en el contexto de una aplicación. Por ejemplo, el estado de la implementación, CloudWatch las alarmas, las configuraciones de los recursos y los problemas operativos.

En la siguiente figura se muestra un ejemplo de la vista de la aplicación para la pila de soluciones de Application Manager.

Muestra la pila de soluciones en Application Manager



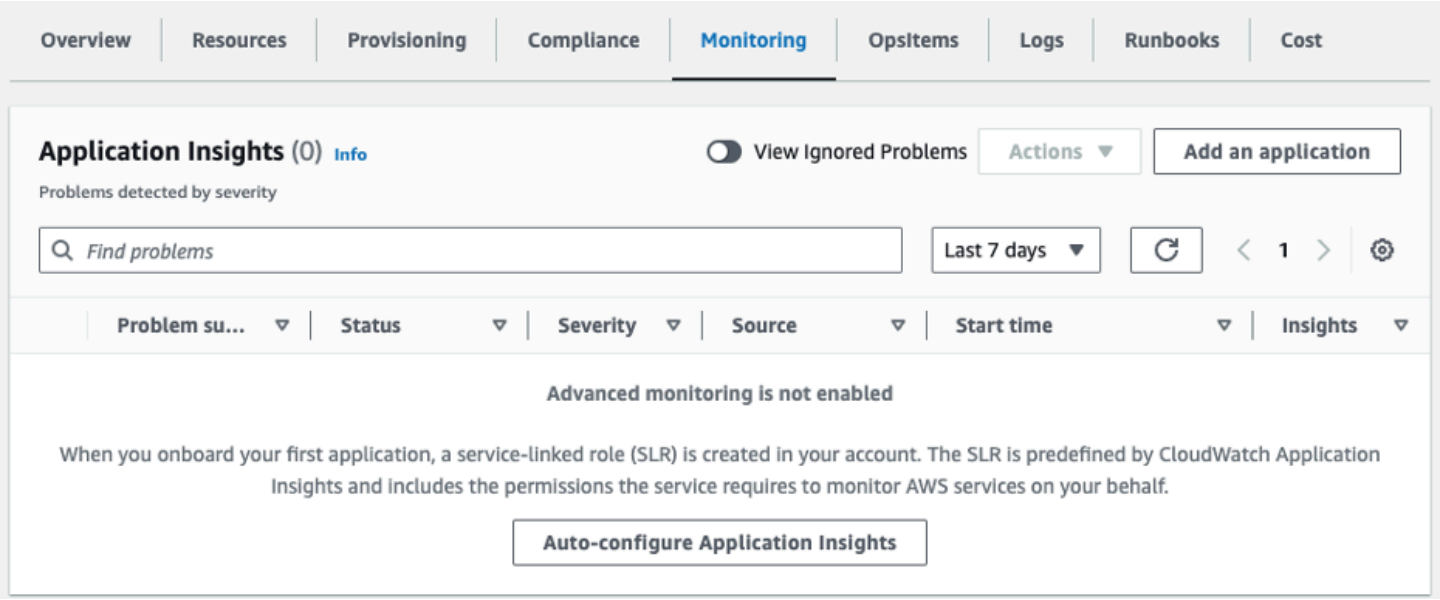
Active Application Insights CloudWatch

1. Inicie sesión en la consola de Administrador de aplicaciones.

2. En el panel de navegación, elija Administrador de aplicaciones.
3. En Aplicaciones, busque el nombre de la aplicación para esta solución y selecciónela.

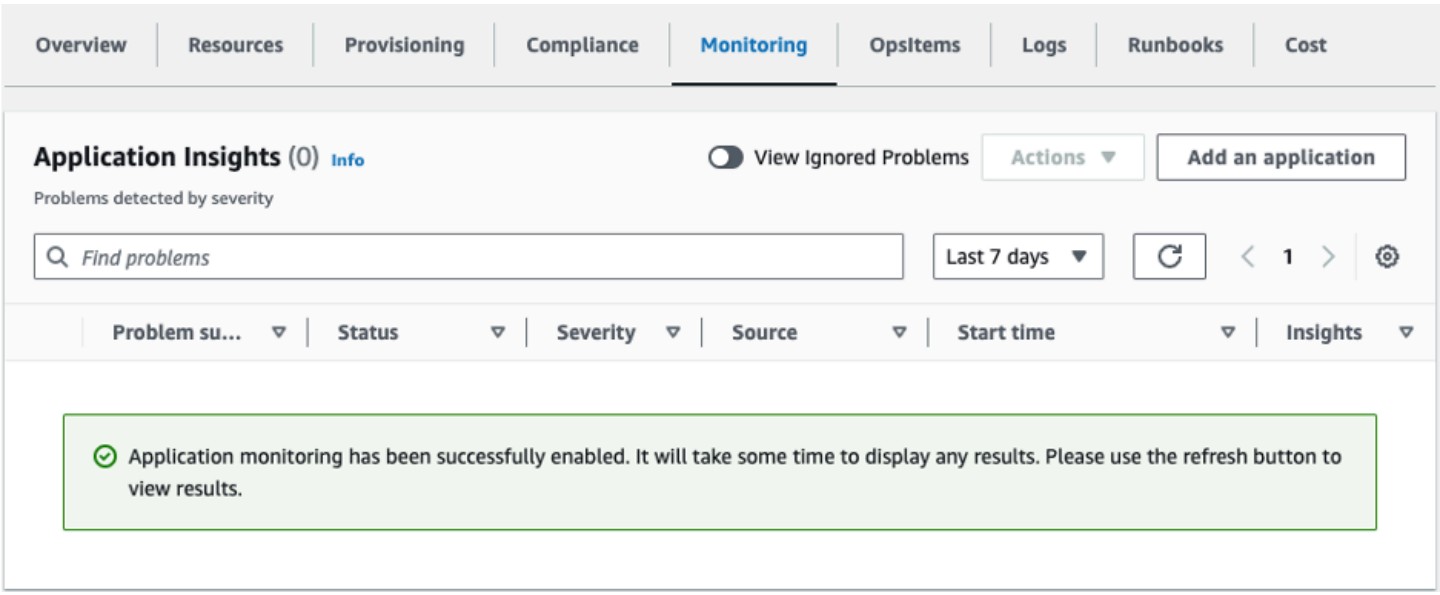
   El nombre de la aplicación tendrá el registro de aplicaciones en la columna Fuente de la aplicación y tendrá una combinación del nombre de la solución, la región, el identificador de cuenta o el nombre de la pila.
4. En el árbol de componentes, elija la pila de aplicaciones que desee activar.
5. En la pestaña Supervisión, en Application Insights, seleccione Configurar automáticamente Application Insights.

El panel de información de aplicaciones muestra los problemas no detectados y la opción de configuración automática.



Ahora, al estar activada la supervisión de sus aplicaciones, aparece el siguiente cuadro de estado:

El panel de Application Insights muestra un mensaje de activación de la supervisión correcta.



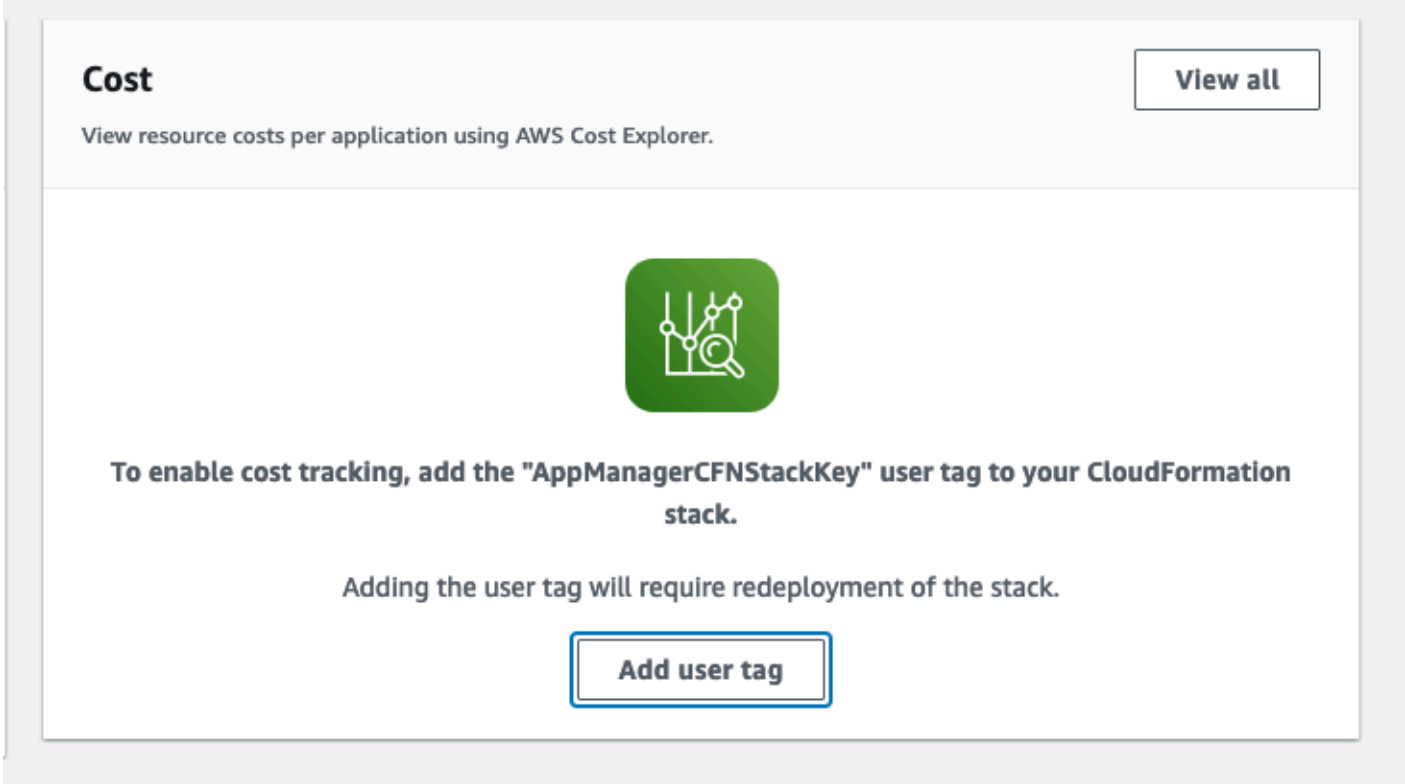
Confirmación de las etiquetas de costos asociadas a la solución

Después de activar Cost Explorer, debe activar las etiquetas de asignación de costos asociadas a esta solución para ver los costos de la solución. Para confirmar las etiquetas de asignación de costos:

1. Inicie sesión en la consola de Administrador de aplicaciones.
2. En el panel de navegación, elija Administrador de aplicaciones.
3. En Aplicaciones, busque el nombre de la aplicación para esta solución y selecciónela.

   El nombre de la aplicación tendrá el registro de aplicaciones en la columna Fuente de la aplicación y tendrá una combinación del nombre de la solución, la región, el identificador de cuenta o el nombre de la pila.
4. En la pestaña Descripción general, en Costo, seleccione Agregar etiqueta de usuario.

   Captura de pantalla que muestra el coste de la aplicación para añadir etiquetas de usuario



5. En la página Agregar etiqueta de usuario, escriba `confirm` y, a continuación, seleccione Agregar etiqueta de usuario.

El proceso de activación puede tardar hasta 24 horas en completarse y en aparecer los datos de la etiqueta.

Activar las etiquetas de asignación de costos asociadas a la solución

Después de activar Cost Explorer, debe activar las etiquetas de asignación de costos asociadas a esta solución para ver los costos de la solución. Las etiquetas de asignación de costos sólo se pueden activar desde la cuenta de administración de la organización. Para activar las etiquetas de asignación de costos:

1. Inicie sesión en la consola de AWS Billing and Cost Management y Cost Management.
2. En el panel de navegación, seleccione Etiquetas de asignación de costos.
3. En la página de etiquetas de asignación de costes, filtre la etiqueta AppManager CFNStack clave y, a continuación, seleccione la etiqueta entre los resultados que se muestran.

4. Seleccione Activar.

Explorador de costos de AWS

Puede ver un resumen de los costes asociados a la aplicación y a los componentes de la aplicación en la consola de Application Manager mediante la integración con AWS Cost Explorer, que debe activarse primero. Cost Explorer le ayuda a administrar los costos al proporcionarle una visión de los costos y el uso de los recursos de AWS a lo largo del tiempo. Para activar Cost Explorer para la solución:

1. Inicie sesión en la consola de administración de costos de AWS.
2. En el panel de navegación, seleccione Cost Explorer para ver los costos y el uso de la solución a lo largo del tiempo.

Actualización de la solución

Si ya implementó la solución anteriormente, siga este procedimiento para actualizar el CloudFormation conjunto de soluciones y obtener las funciones y mejoras más recientes. El proceso de actualización consta de tres partes:

- Paso 1: Actualizar el panel de implementación
- Paso 2: Migrar las configuraciones de los casos de uso
- Paso 3: Actualizar los casos de uso

> **Note**
>
> 1. En la versión 2.0.0, la integración con Anthropic y Hugging Face quedó obsoleta en favor de Amazon Bedrock y Amazon AI. SageMaker Puedes implementar los modelos disponibles a través de Hugging Face SageMaker JumpStart hasta. Consulta Cómo usar Hugging Face con SageMaker Amazon AI para obtener más información.
> 2. Asegúrese de probar el proceso de actualización en un entorno que no sea de producción antes de ejecutar estos pasos.

Paso 1: Actualizar el panel de implementación

1. Inicia sesión en la CloudFormation consola, selecciona tu CloudFormation pila actual y selecciona Actualizar.
2. Seleccione Reemplazar la plantilla actual.
3. En Especificar plantilla:
   a. Seleccione URL de Amazon S3.
   b. Copia el enlace de la CloudFormation plantilla más reciente.
   c. Pegue el enlace en el cuadro URL de Amazon S3.
   d. Verifique que la URL de la plantilla correcta aparezca en el cuadro de texto URL de Amazon S3 y seleccione Siguiente. Vuelva a seleccionar Siguiente.

4. En Parámetros, revise los parámetros de la plantilla y modifíquelos según sea necesario. Para obtener más información sobre los parámetros, consulte el paso 1: lanzar la pila de paneles de despliegue.
5. Elija Siguiente.
6. En la página Configurar opciones de pila, elija Siguiente.
7. En la página Revisar, revise y confirme la configuración. Seleccione la casilla para reconocer que la plantilla creará recursos de IAM.
8. Seleccione Ver conjunto de cambios y verifique los cambios.
9. Seleccione Crear pila para implementar la pila.

Puede ver el estado de la pila en la CloudFormation consola de AWS en la columna Estado. Debería recibir el estado UPDATE_COMPLETE en aproximadamente 10 minutos.

Si la versión de la solución existente era anterior a la v2.0.0, la actualización creará una pila de interfaz de usuario web (que sustituye la `amplify-ui` implementación de la pantalla de inicio de sesión por una interfaz de usuario alojada en Cognito) y una nueva CloudFront URL, que se puede obtener en la sección de resultados de la CloudFormation consola una vez que el estado de la pila sea UPDATE_COMPLETE.

> **Note**
>
> Los casos de uso existentes creados con versiones anteriores a la v2.0.0 no se mostrarán hasta que complete los pasos que se describen a continuación.

Paso 2: Migrar las configuraciones de los casos de uso (solo las actualizaciones de versiones anteriores a la 2.0.0)

El esquema de almacenamiento y la configuración de los casos de uso del servicio de AWS para almacenar cambiaron en la versión 2.0.0. Siga los pasos descritos en la Guía del usuario de migración a GAAB v2 con el script gaab_v2_migration.py. Tras ejecutar el script, puede acceder al panel de implementación para ver los casos de uso implementados.

> **Note**
>
> Debe seguir los pasos que se indican a continuación para completar la migración de los casos de uso.

Paso 3: Actualizar los casos de uso

Puede editar los casos de uso implementados con las nuevas funciones disponibles en las últimas versiones de la GAAB. Consulte Usar la solución para obtener información sobre cómo usar las funciones de esta solución.

Para actualizar los casos de uso a la última versión, debe completar los pasos de «Editar casos de uso» del panel de implementación (aunque es posible que no realice ningún cambio). Esta acción desencadena una actualización de la CloudFormation pila con la última versión de la plantilla.

> **Note**
>
> Es posible que los casos de uso creados con las versiones 1.x o 2.x de la solución no funcionen con versiones posteriores. Por lo tanto, recomendamos clonar los casos de uso existentes creados con versiones anteriores a la v3.0.0 a través del panel de implementación. A continuación, migre gradualmente y sustitúyalos por nuevos casos de uso creados con la versión 3.0.0 o una versión posterior.

Resolución de problemas

En esta sección se proporcionan instrucciones de solución de problemas para la implementación y el uso de la solución.

Si estas instrucciones no abordan su problema, el artículo de Contacto con soporte proporciona instrucciones que le ayudan a abrir un caso de soporte para esta solución.

Problema: la implementación de una configuración habilitada para VPC, con Create a VPC for me, falla

La pila de paneles de implementación o la pila de casos de uso fallan en la CloudFormation implementación porque no pudieron aprovisionar los recursos de red de la VPC.

Resolución

Comprueba los límites de cuota de VPCs Elastic y los IPs de tu cuenta. Los límites predeterminados son 5 para Elastic IPs y VPCs 5 para cada cuenta de AWS, por región de AWS.

> **Note**
>
> Cuando la solución crea una VPC, una sola implementación habilitada para VPC (panel de implementación o caso de uso) es una implementación en 2 zonas de disponibilidad con 1 subred pública y 1 subred privada en cada zona de disponibilidad, y cada subred pública implementa 1 puerta de enlace NAT. Con 2 pasarelas NAT, la implementación consume 2 direcciones IP públicas del límite de cuota.

Algunos límites que hay que tener en cuenta (por cuenta, por región):

- Número de VPCs : 5
- Número de direcciones IP públicas: 5
- Número de puntos finales de VPC de puerta de enlace: 20
- Número de puntos finales de VPC de interfaz: 20

Problema: la pila de casos de uso no se puede eliminar una CloudFormation vez eliminada la pila del panel de implementación

Si se elimina la pila del panel de implementación CloudFormation antes de eliminar todas las pilas de casos de uso, los casos de uso pueden terminar bloqueados (inutilizables). Esto se debe a que una función de IAM creada por la pila de paneles de despliegue ya no existe, lo que impide modificar la pila de casos de uso.

Resolución

> **Warning**
>
> Asegúrese de limpiar todos los roles creados manualmente inmediatamente después de usarlos. Se trata de permisos elevados que los usuarios podrían aprovechar para elevar los roles.

Vuelva a crear el rol de IAM eliminado para permitir la eliminación de las CloudFormation pilas:

1. Abre la CloudFormation consola y determina la función asociada a la pila bloqueada.
   a. El ARN del rol se encuentra en la sección de información de la pila denominada rol de IAM.
   b. El nombre del rol es el que sigue después de:role/ en el ARN del rol de IAM (por ejemplo, arn:aws:iam: :role/) <account-id><role-name>
2. Cree un nuevo rol en IAM con el mismo nombre que el rol eliminado.
   a. Seleccione el servicio AWS como entidad de confianza y selecciónelo CloudFormationen el menú desplegable.
   b. Añada los permisos necesarios. Si no está seguro de los permisos necesarios, puede utilizar la AdministratorAccesspolítica gestionada de AWS.
   c. Introduzca el nombre del rol exactamente como se indica en el paso 1.
3. Regrese a la CloudFormation consola y elimine las pilas bloqueadas.
4. Una vez que se hayan eliminado correctamente todas las pilas bloqueadas, vuelva a IAM y elimine todas las funciones creadas en el paso 2.

Problema: la interfaz de usuario del caso de uso no refleja los cambios en la configuración

Cuando se actualizan los casos de uso, la interfaz de usuario se despliega en CloudFront. Sin embargo, dado que almacena en CloudFront caché las implementaciones y el archivo de configuración que dicta cómo se muestran algunos ajustes al usuario, es posible que estos cambios no se reflejen de forma inmediata.

Resolución

La CloudFront distribución se puede invalidar para forzar la propagación de la nueva configuración a los usuarios de front-end.

1. Abra la CloudFormation consola y determine la CloudFront distribución asociada a su conjunto de casos de uso.
   a. La pila de casos de uso debe empezar con el mismo nombre que utilizó al implementar el caso de uso.
   b. Localice la pila anidada correspondiente a la interfaz de usuario. El nombre de la pila anidada debe empezar por S3 WebAppStackS3 UINested. UINested StackResource
   c. En la pestaña Recursos, localice el tipo de recurso y, a continuación AWS::CloudFront::Distribution, seleccione el ID físico. Esto abrirá la distribución en la CloudFront consola.
2. Vaya a la pestaña Invalidaciones y, a continuación, seleccione Crear invalidación e introduzca la ruta /*. Esto invalidará todas las rutas.
3. En su propio navegador, elimine las cookies y los archivos en caché relacionados con el caso de uso.

Póngase en contacto con AWS Support.

Si tiene AWS Business Support+, AWS Enterprise Support o Unified Operations, puede utilizar el AWS Support Center para obtener asistencia de expertos con esta solución. En las siguientes secciones, encontrará instrucciones.

Crear caso

1. Inicie sesión y vaya al Centro de soporte.

2. Seleccione Crear caso.

¿Cómo podemos ayudarle?

1. Elija Técnico.
2. En Servicio, seleccione Soluciones.
3. En Categoría, seleccione Otras soluciones.
4. En Gravedad, seleccione la opción que mejor se adapte a su caso de uso.
5. Al especificar los valores de Servicio, Categoría y Gravedad, la interfaz rellena los enlaces a las preguntas más frecuentes de solución de problemas. Si no puede resolver su pregunta con estos enlaces, elija Paso siguiente: información adicional.

Información adicional

1. En Asunto, introduzca un texto que resuma su pregunta o problema.
2. Para obtener una descripción, describa el problema en detalle, incluido el nombre de esta solución: Generative AI Application Builder en AWS.
3. Elija Adjuntar archivos.
4. Adjunte la información que AWS Support necesita para procesar la solicitud.

Ayúdenos a resolver su caso más rápido

1. Especifique la información requerida.
2. Elija Siguiente paso: Resuelva ahora o póngase en contacto con nosotros.

Resuelva ahora o póngase en contacto con nosotros

1. Revise las soluciones de Resolver ahora.
2. Si estas no le ayudan a resolver su problema, elija Contactar con nosotros, especifique la información solicitada y seleccione Enviar.

Desinstalar la solución

> **Note**
>
> Las implementaciones creadas a través del panel de implementación no están diseñadas para administrarse fuera de la solución. Asegúrese de eliminar y limpiar todas las implementaciones desde el panel de implementación antes de eliminar la pila. CloudFormation

Puede desinstalar la solución Generative AI Application Builder on AWS desde la consola de administración de AWS o mediante la interfaz de línea de comandos de AWS. Debe eliminar manualmente los buckets de Amazon S3, los índices de Amazon Kendra CloudWatch o los registros creados por esta solución. Las soluciones de AWS no eliminan automáticamente los buckets, los índices de Amazon Kendra CloudWatch o los registros de Amazon S3 en caso de que haya almacenado datos para conservarlos.

Uso de Consola de administración de AWS

1. Inicie sesión en la CloudFormation consola de AWS.
2. En la página Pilas, seleccione la pila de instalación de esta solución.
3. Elija Eliminar.

Uso de la Interfaz de la línea de comandos de AWS

Determine si la Interfaz de la línea de comandos de AWS (AWS CLI) está disponible en su entorno. Para obtener instrucciones de instalación, consulte Qué es la interfaz de línea de comandos de AWS en la Guía del usuario de la CLI de AWS. Tras confirmar que la CLI de AWS está disponible, ejecute el siguiente comando:

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

Pasos de desinstalación manual

Eliminar los buckets de Amazon S3

Esta solución está configurada para conservar el bucket de Amazon S3 creado por la solución si decide eliminar la CloudFormation pila de AWS para evitar la pérdida accidental de datos. Tras desinstalar la solución, puede eliminar manualmente este bucket de Amazon S3 si no necesita conservar los datos. Siga estos pasos para eliminar el bucket de Amazon S3.

1. Inicie sesión en la consola de Amazon S3.
2. En el panel de navegación, selecciona Buckets.
3. Localice los depósitos <stack-name>S3.
4. Seleccione el depósito de S3 y elija Eliminar.

Para eliminar el bucket de S3 mediante la AWS CLI, ejecute el siguiente comando. No necesitará vaciar primero el depósito cuando utilice la opción --force.

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

Eliminar los índices de Amazon Kendra

Para evitar la pérdida accidental de datos, esta solución está configurada para conservar los índices de Amazon Kendra creados por la solución cuando se elimine la pila de CloudFormation AWS. Tras desinstalar la solución, puede eliminar manualmente los índices de Amazon Kendra para los que ya no necesita conservar datos. Sigue estos pasos para eliminar el índice de Amazon Kendra.

1. Inicia sesión en la consola de Amazon Kendra.
2. En el panel de navegación, seleccione Índices.
3. Busque y seleccione el índice que desee eliminar.
4. Elija Eliminar para eliminar el índice seleccionado.

Para eliminar el índice de Amazon Kendra mediante la AWS CLI, ejecute el siguiente comando:

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

Eliminar los registros CloudWatch

Para evitar la pérdida accidental de datos, configuramos esta solución para conservar los CloudWatch registros si decide eliminar la CloudFormation pila. Tras desinstalar la solución, puede eliminar los registros manualmente si no necesita conservar los datos. Siga estos pasos para eliminar los CloudWatch registros.

1. Inicia sesión en la CloudWatch consola de Amazon.
2. En el panel de navegación, selecciona Grupos de registro.
3. Localice los grupos de registros creados por la solución.
4. Seleccione uno de los grupos de registros.
5. Elija Acciones y, a continuación, elija Eliminar.

Repita los pasos hasta que haya eliminado todos los grupos de registros de soluciones.

Uso de la solución

Acceso a la interfaz de usuario

Durante el proceso de implementación de la pila (tanto para el panel de implementación como para los casos de uso), se envía un correo electrónico a la dirección de correo electrónico configurada. El correo electrónico contiene las credenciales temporales del usuario que puede usar para registrarse y acceder a la interfaz web.

> **Note**
>
> El DevOps usuario con acceso a la consola de administración de AWS debe proporcionar al usuario administrador la CloudFront URL de la interfaz de usuario del panel de implementación cuando se complete la pila.

Para los casos de uso, el usuario administrador con acceso a la interfaz de usuario del panel de implementación debe proporcionar al usuario empresarial la CloudFront URL de la interfaz de usuario del caso de uso cuando se complete la implementación.

Una vez que haya iniciado sesión, el usuario puede interactuar con la solución UIs, ya sea en el panel de implementación en el caso de los administradores o en el caso de uso en el caso de los usuarios empresariales.

¿Cómo actualizar una implementación

En la página de inicio del panel de control de despliegue (o en la página de detalles de un despliegue), puede editar la configuración utilizada por un despliegue. Solo puede editar las implementaciones que se encuentren en los estados CREATE_COMPLETE o UPDATE_COMPLETE.

A excepción del nombre del caso de uso, todas las demás opciones se pueden editar para una implementación. Solo tiene que cambiar los valores que desee editar y volver a implementar.

Según el alcance de las ediciones realizadas, el tiempo de redistribución variará. Pueden pasar unos segundos si se ha modificado una configuración simple (por ejemplo, los parámetros del modelo) o

más de 30 minutos si se han modificado opciones más amplias relacionadas con la infraestructura (por ejemplo, solicitar la creación del índice de Amazon Kendra para el caso de uso de texto RAG).

Una vez que la edición se haya completado correctamente, el estado de la solicitud mostrará el estado UPDATE_COMPLETE. En este momento, puede acceder a la interfaz de usuario implementada a través de la CloudFront URL e interactuar con la implementación modificada.

> **Note**
>
> Puede que sea más fácil ejecutar varias implementaciones side-by-side si desea comparar diferentes configuraciones o LLMs. Utilice la función de clonación para utilizar rápidamente una configuración existente para lanzar una nueva implementación.

¿Cómo clonar una implementación

En la página de inicio del panel de control de despliegues (o en la página de detalles de un despliegue), puede clonar la configuración utilizada por un despliegue. Al clonar una implementación, se inicia el asistente Implementar nuevos casos de uso, pero la mayoría de los campos se rellenan previamente con los mismos valores.

Se trata de una práctica operación que le ayuda a duplicar rápidamente despliegues con una configuración modificada, reactivar un despliegue eliminado o comparar varios despliegues que, por lo demás, serían LLMs idénticos.

¿Cómo eliminar una implementación

Cuando se encuentre en la página de inicio del panel de control de despliegues (o en la página de detalles de un despliegue), podrá eliminarlo cuando ya no necesite el despliegue. Al eliminar una implementación, se invoca una operación de eliminación de CloudFormation pilas y se desaprovisionan los recursos para la implementación.

De forma predeterminada, una implementación eliminada permanece en el panel de control para habilitar la funcionalidad de clonación. Para eliminar por completo una implementación del panel de control y dejar de rastrearla en la interfaz de usuario, selecciona Eliminar permanentemente en la ventana de confirmación de eliminación.

> **Important**
>
> Algunos recursos se quedan atrás durante la eliminación de la pila y se deben eliminar manualmente. Consulte la sección de desinstalación manual para obtener más información sobre los recursos que se conservan y cómo limpiarlos.

Configuración de un modelo de lenguaje grande (LLM)

El LLM adecuado para su caso de uso depende de un amplio conjunto de factores específicos de sus necesidades y del tipo de experiencia de cliente que desee personalizar. Esta solución no pretende ser prescriptiva, sino que pretende proporcionarle las herramientas necesarias para evaluar qué es lo que mejor se adapta a su aplicación.

El espacio generado por la IA está evolucionando rápidamente, por lo que es su responsabilidad mantenerse al día con los últimos modelos, técnicas de optimización y mejores prácticas para garantizar que está creando las experiencias adecuadas para sus clientes.

> **Note**
>
> Si trabaja con datos confidenciales o privados, asegúrese de seleccionar una opción de LLM con los servicios de AWS (como Amazon Bedrock o Amazon SageMaker AI). Esto mejora la seguridad general de su implementación al mantener los datos dentro de su región y en la red de AWS en comparación con el uso de un LLM alojado por un proveedor externo.

Uso de Amazon SageMaker AI como proveedor de LLM

A partir de la versión 1.3.0, Amazon SageMaker AI está disponible como proveedor de modelos para casos de uso de texto. Esta función le permite utilizar un punto final de inferencia de SageMaker IA que ya existe en la cuenta de AWS en la solución. Estas son algunas formas de empezar.

> **Important**
>
> La solución no gestiona el ciclo de vida de sus terminales de SageMaker IA. Usted es responsable de eliminar los puntos finales de SageMaker IA una vez que ya no sean necesarios para dejar de incurrir en cargos adicionales.

Crear un punto final de IA SageMaker

Puede usar Amazon SageMaker AI JumpStart para implementar rápidamente un punto final.

También puede utilizar un punto final de SageMaker IA basado en la generación de texto e implementarlo mediante el servicio de SageMaker IA básico. Consulte la JumpStart documentación de SageMaker IA para obtener una guía paso a paso sobre cómo implementar un modelo de inferencia.

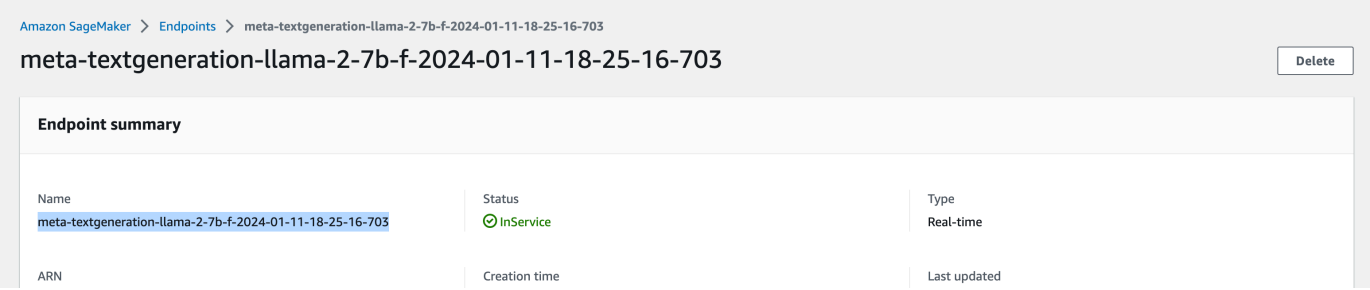
> **Note**
>
> models/LLMs Las bases suelen ser bastante grandes y, a menudo, pueden requerir el uso de grandes instancias de cómputo acelerado. Es posible que muchas de estas instancias más grandes no estén disponibles de forma predeterminada en su cuenta de AWS. Consulte las cuotas de SageMaker IA predeterminadas y asegúrese de solicitar un aumento de cuota antes de realizar la implementación para evitar errores habituales en la implementación.

Utilice un punto final de SageMaker IA para crear un despliegue de casos de uso de Text

Para implementar un nuevo caso de uso de Text utilizando un punto final de SageMaker IA a modo de inferencia:

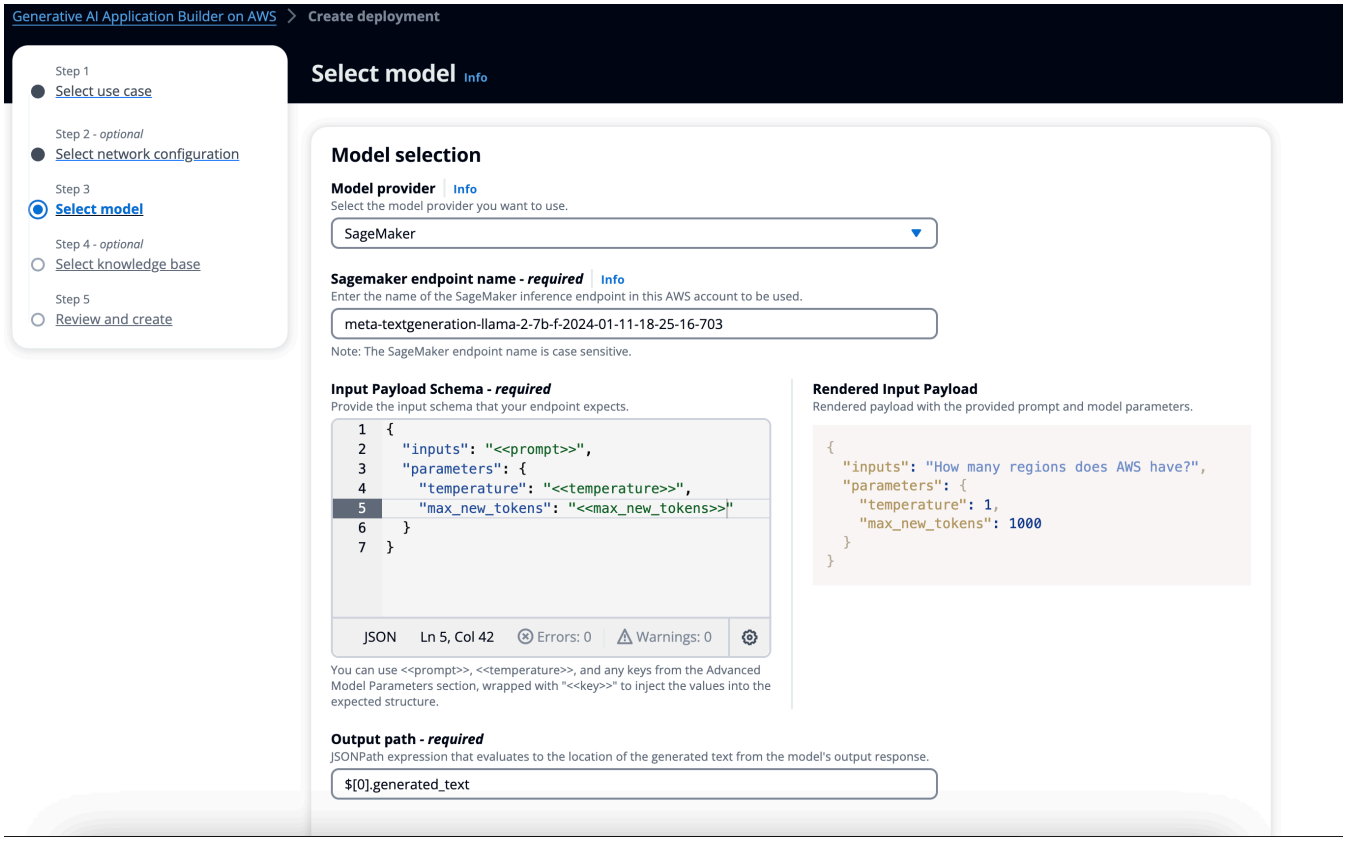
1. Cree un nuevo caso de uso mediante el asistente del panel de implementación y complete los formularios hasta llegar a la página de selección de modelos.
2. En la página de modelos, seleccione SageMaker AI como proveedor de modelos. Esto generará un formulario personalizado que requerirá tres entradas clave del usuario:
   - El nombre del punto final de SageMaker IA que quieres usar. DevOps los usuarios pueden obtenerlo desde la consola de AWS. Tenga en cuenta que el punto de conexión debe estar en la misma cuenta y región en las que está implementada la solución.

     Ubicación del nombre del punto de conexión en la consola de AWS



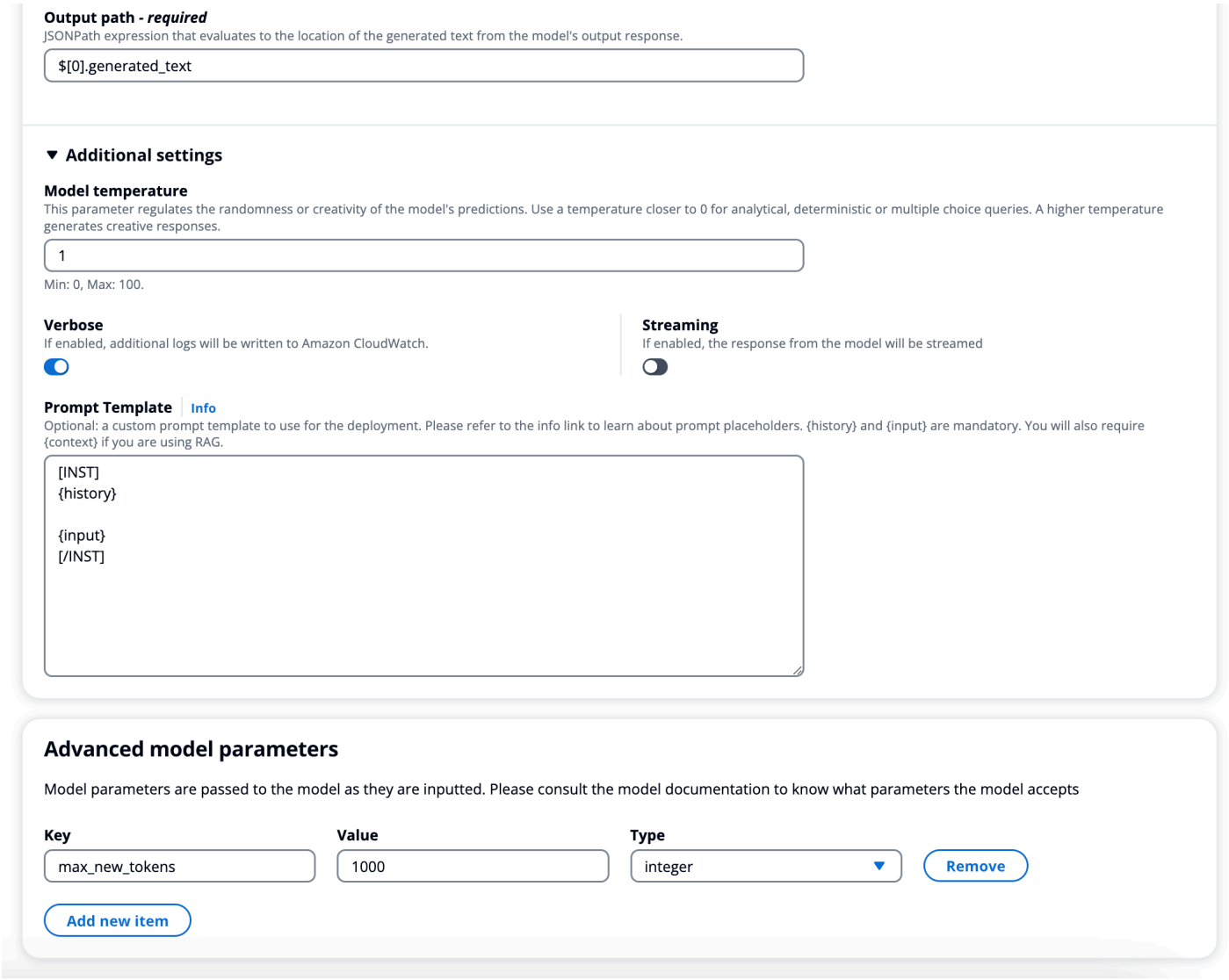
- El esquema de la carga útil de entrada que espera el punto final. Para admitir el conjunto más amplio de puntos finales, los usuarios administradores deben indicar a la solución cómo espera su punto final que se formatee la entrada. En el asistente de selección de modelos, proporcione el esquema JSON para que la solución se envíe al punto final. Puede añadir marcadores de posición para introducir valores estáticos y dinámicos en la carga útil de la solicitud. Las opciones disponibles son:
  - Los marcadores de posición obligatorios:\ <\ <prompt\ >\ > se sustituirán dinámicamente por la entrada completa (por ejemplo, el historial, el contexto y la entrada del usuario según la plantilla de solicitud) y se enviarán al punto final de la SageMaker IA durante el tiempo de ejecución.
  - Se <temperature\ > pueden proporcionar al punto final marcadores de posición opcionales: \ <\\ > *,\ *, así como cualquier parámetro definido en los parámetros avanzados del modelo. Cualquier cadena que contenga un marcador de posición entre\ <\ < and\ >\ > (por ejemplo,\ <\ <max_new_tokens\ >\ >) se sustituirá por el valor del parámetro del modelo avanzado del mismo nombre.

    Ejemplo de esquema de entrada: configuración de campos obligatorios, indicador y temperatura, junto con un parámetro avanzado personalizado, max_new_tokens. La ruta de salida debe proporcionarse como una cadena válida JSONPath



3. La ubicación de la respuesta de cadena LLMs generada dentro de la carga útil de salida. Debe proporcionarse como una JSONPath expresión para indicar desde dónde se espera acceder a la respuesta de texto final que se muestra a los usuarios desde el objeto devuelto y la respuesta del punto final.

   Ejemplo de cómo añadir parámetros de modelo avanzados para utilizarlos en un esquema de entrada de SageMaker IA (consulte la figura 2 para ver las opciones y ajustes anteriores)



> **Note**
>
> SageMaker La IA ahora admite el alojamiento de varios modelos en el mismo punto final, y esta es la configuración predeterminada al implementar un punto final en la versión actual de SageMaker AI Studio (no en Studio Classic).
> Si tu terminal está configurado de esta manera, tendrás que añadirlo InferenceComponentNamea la sección de parámetros avanzados del modelo con un valor que corresponda al nombre del modelo que quieras usar.

Configuración avanzada de LLM

Al utilizar Amazon Bedrock, puede configurar algunos ajustes avanzados para sus modelos, como Amazon Bedrock Guardrails, el rendimiento aprovisionado para Amazon Bedrock y parámetros de modelo adicionales.

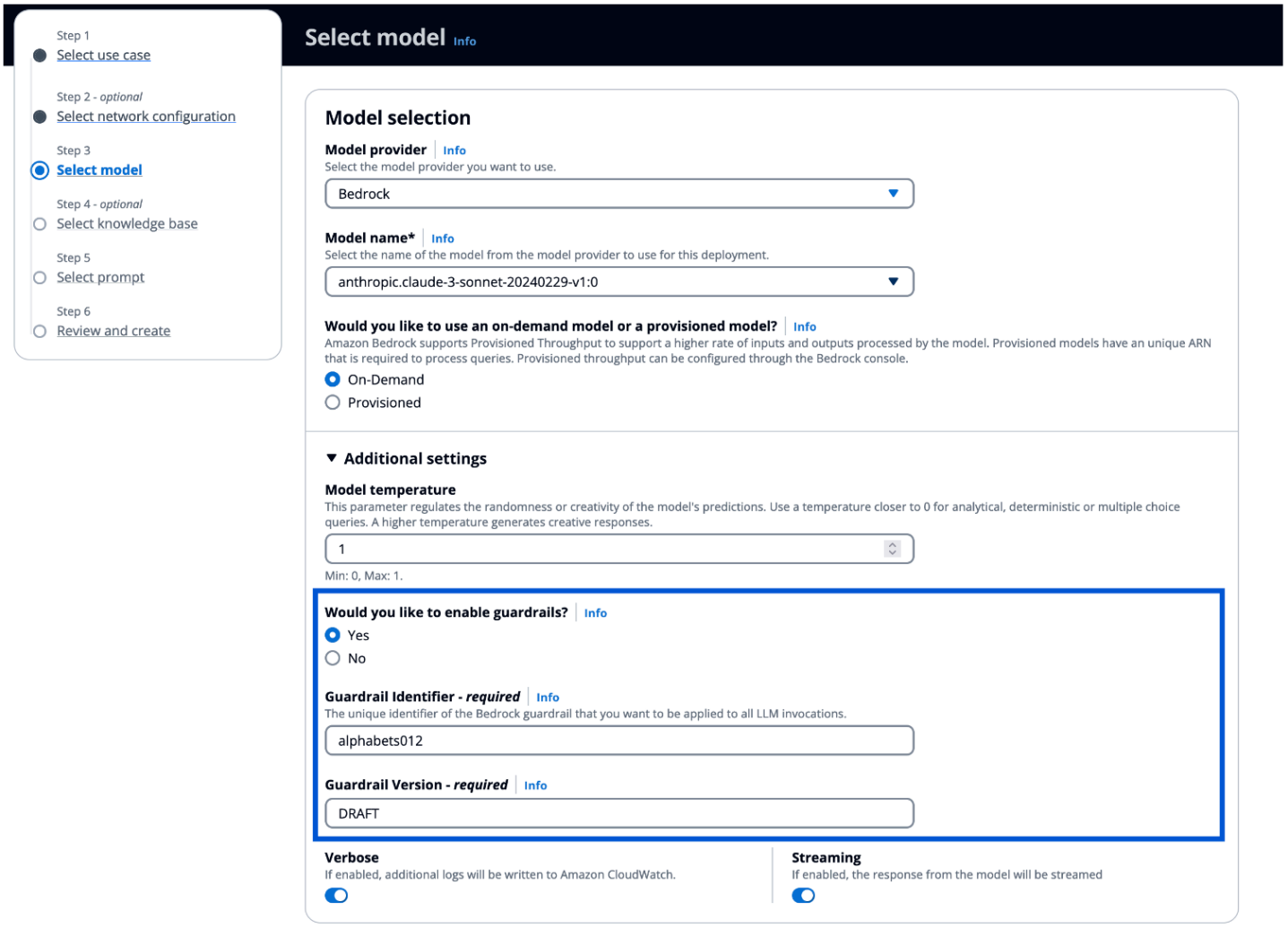
Barreras de protección para Amazon Bedrock

Amazon Bedrock Guardrails es una función de Amazon Bedrock que evalúa las entradas de los usuarios y las respuestas de LLM en función de las políticas configuradas por el usuario y proporciona un nivel adicional de protección, independientemente del LLM subyacente que el usuario seleccione para un caso de uso. Un Guardrail consta de dos políticas para evitar que el contenido se clasifique en las categorías no deseadas o dañinas:

1. Temas denegados para definir un conjunto de temas que no son deseables en el contexto de la solicitud del usuario, por ejemplo, el asesoramiento de inversión en una aplicación financiera y,
2. Filtros de contenido****que permiten filtrar los mensajes introducidos por los usuarios o modelar las respuestas que contienen contenido dañino.

Para su uso en la solución Generative AI Application Builder, se debe configurar una barandilla en la consola de Amazon Bedrock mediante el asistente Create Guardrail. Una vez creado, puede añadir este guardrail a su caso de uso de chat creado mediante el asistente de la solución Generative AI Application Builder en los ajustes adicionales del paso de selección del modelo, proporcionando su identificador de barandilla y la versión de guardarail.

Muestra el asistente de implementación, que habilita Amazon Bedrock Guardrails



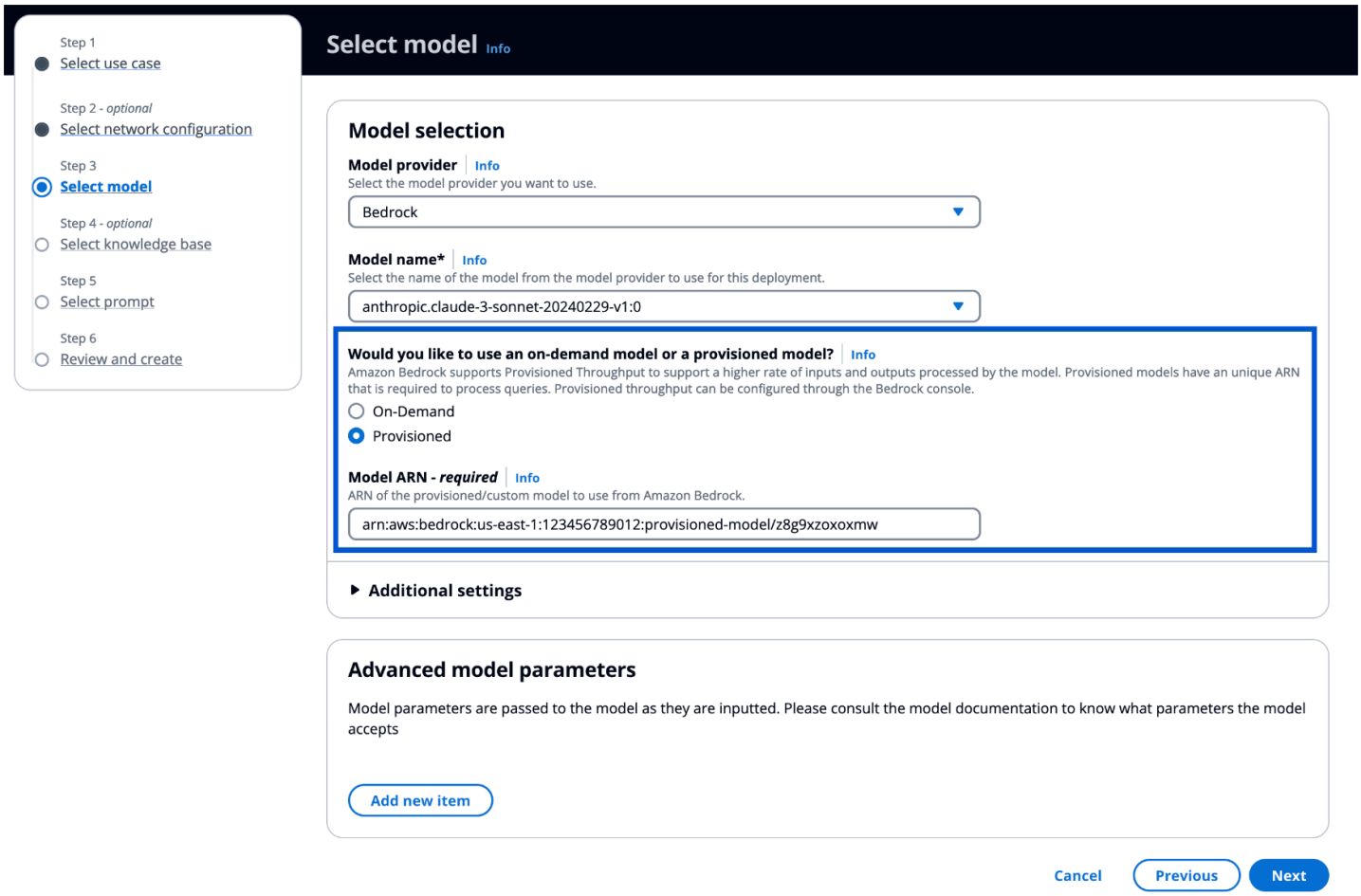
Rendimiento aprovisionado para Amazon Bedrock

Cada modelo de Amazon Bedrock bajo demanda sigue el límite de cuota de cuenta específico de la región para la inferencia de modelos. Por ejemplo, Anthropic Claude 2.x en Bedrock actualmente permite procesar 500 solicitudes y 500 000 fichas por minuto en las regiones us-east-1 y us-west-2. Es posible que también desee utilizar la solución con sus modelos perfeccionados o previamente entrenados de forma continua. En estos casos, Amazon Bedrock permite un rendimiento aprovisionado, lo que permite ejecutar grandes cargas de trabajo de inferencias consistentes para su base, modelos ajustados o continuos previamente entrenados para su uso en aplicaciones de nivel de producción.

Una vez que se adquiere el rendimiento aprovisionado en la consola Amazon Bedrock, se genera un ARN modelo para su uso. Ahora puede proporcionar este ARN de modelo en el asistente Generative AI Application Builder en el paso de selección del modelo. Para ello, seleccione Bedrock

como proveedor del modelo y el nombre del modelo base que se utilizó para generar este ARN de modelo aprovisionado en la consola de Amazon Bedrock. A continuación, seleccione «Modelo aprovisionado» al elegir entre los modelos bajo demanda y aprovisionados y suministre el ARN de su modelo.

Describe el asistente de implementación: Cómo habilitar el rendimiento aprovisionado para Amazon Bedrock



> **Note**
>
> El rendimiento provisional y el aprovisionamiento deben estar en la misma región que el panel de implementación implementado y las pilas de casos de uso.

Parámetros del modelo

LLMs suelen aceptar una amplia gama de parámetros específicos para su implementación. Los proveedores de modelos suelen proporcionar documentación en la que se describe el conjunto de parámetros admitidos y sus usos.

La solución transfiere los parámetros del modelo directamente al modelo subyacente, por lo que es importante asegurarse de que los parámetros estén configurados correctamente. Consulte la documentación del proveedor del modelo para obtener la información más reciente sobre los parámetros admitidos.

Configuración de Agent Builder

Agent Builder ofrece opciones de configuración completas para crear agentes de IA listos para la producción. En esta sección se describe cómo configurar y gestionar las implementaciones de Agent Builder.

Configuración rápida del sistema

El mensaje del sistema define el comportamiento, la personalidad y las capacidades de su agente. Para configurar la línea de comandos del sistema:

1. En el asistente Agent Builder, vaya al paso Configurar el agente.
2. Edite la plantilla de solicitud del sistema en el editor de texto.
3. Incluya instrucciones claras para:
   - Función y propósito del agente
   - Cómo utilizar las herramientas disponibles (servidores MCP)
   - Preferencias de formato de respuesta
   - Pautas de comportamiento
4. Utilice el botón Restablecer valores predeterminados para restaurar la plantilla original si es necesario.

Mejores prácticas para las solicitudes de los agentes:

- Sea específico en cuanto a las capacidades y limitaciones del agente
- Proporcione ejemplos claros del comportamiento deseado

- Incluya instrucciones sobre el uso de la herramienta y cuándo invocarlas
- Defina las expectativas del formato de respuesta
- Establezca límites para el comportamiento de los agentes

Integración de servidores MCP

Los servidores del Protocolo de Contexto Modelo (MCP) proporcionan a los agentes acceso a las herramientas empresariales y a las fuentes de datos. Para configurar los servidores MCP:

1. En el paso Configurar el agente, busque la sección Servidores MCP.
2. Seleccione uno de los servidores MCP disponibles en el menú desplegable.

> **Note**
>
> Los servidores MCP deben estar configurados y ser accesibles antes del despliegue del agente. El agente descubrirá y utilizará automáticamente las herramientas expuestas por los servidores MCP configurados. Consulte la documentación del MCP para obtener información sobre la configuración del servidor y la configuración de las herramientas.

Configuración de memoria

Agent Builder proporciona dos tipos de memoria para mantener el contexto y el conocimiento:

Memoria a corto plazo

Habilitado de forma predeterminada para todos los agentes:

- Mantiene el contexto de la conversación dentro de las sesiones
- Captura automáticamente los mensajes de los usuarios y las respuestas de los agentes
- Organizado por ActorID y SessionID para un aislamiento adecuado
- No se requiere configuración

Memoria a largo plazo

Función opcional para almacenar información en todas las sesiones:

1. En el paso Configurar el agente, busque la sección Configuración de memoria.
2. Active la opción Habilitar la memoria de larga duración para que se active.
3. Cuando está activado, el agente puede:
   - Extraer y almacenar información importante de las conversaciones
   - Recupera el contexto relevante de sesiones anteriores
   - Desarrolle conocimientos sobre las preferencias y el historial de los usuarios

> **Note**
>
> La memoria a largo plazo utiliza AgentCore la memoria con una estrategia de memoria semántica y una configuración de retención predeterminada.

Supervisión de las implementaciones de Agent Builder

Agent Builder proporciona una supervisión integral a través de CloudWatch paneles y métricas.

Acceso a CloudWatch los paneles

1. Diríjase a la CloudWatch consola de su cuenta de AWS.
2. Seleccione Dashboards en la barra de navegación de la izquierda.
3. Busque el cuadro de mando denominado`AgentBuilder-<UseCaseId>`.
4. Consulta las métricas y los datos históricos de rendimiento en tiempo real.

Acceso y análisis de registros

Los registros de los agentes están disponibles en CloudWatch los registros:

1. Diríjase a CloudWatch Logs en la consola de AWS.
2. Busque grupos de registros con `/aws/bedrock-agentcore/runtimes/` el prefijo.
3. Use CloudWatch Insights para consultar y analizar los registros.
4. Busque patrones de solicitud IDs o error específicos.

Configuración de Workflow Builder

Workflow Builder permite la organización de varios agentes a través de un agente supervisor que delega el trabajo en agentes especializados de Agent Builder.

Creación de un flujo de trabajo

1. Navegue hasta el panel de implementación
2. Seleccione Crear caso de uso de flujo de trabajo
3. Configure el agente supervisor:
   - Nombre: nombre descriptivo del flujo de trabajo
   - Descripción: Propósito y capacidades
   - System Prompt: instrucciones para la delegación y coordinación de los agentes
   - Modelo: modelo básico para el agente supervisor

Mejores prácticas para las indicaciones del supervisor:

- Describa claramente cuándo utilizar cada agente especializado
- Incluya instrucciones para agregar los resultados de varios agentes
- Defina las expectativas de formato de respuesta
- Establezca límites para el comportamiento de las delegaciones

Selección de agentes

Seleccione los agentes de Agent Builder para incluirlos como agentes especializados:

1. Haga clic en Añadir agente en la configuración del flujo de trabajo
2. Examine o busque los agentes de Agent Builder disponibles
3. Revisa las descripciones de los agentes
4. Seleccione los agentes que desee incluir en el flujo de trabajo

Descripciones de agentes

El agente supervisor usa las descripciones de los agentes para decidir en qué agente delegar. Asegúrese de que las descripciones expliquen claramente:

- Dominio o capacidad especializados del agente
- Tipos de tareas que gestiona el agente
- Expectativas de entrada/salida

Probar flujos de trabajo

Tras el despliegue:

1. Acceda al flujo de trabajo a través del panel de implementación
2. Realice pruebas con consultas que requieran varios agentes
3. Supervise la delegación de agentes en CloudWatch los registros
4. Revise la calidad de las respuestas y los patrones de delegación
5. Ajuste el aviso del supervisor si la delegación no es óptima

Consejos para gestionar los límites de los tokens de los modelos

Nota: La solución no intenta gestionar directamente los límites de token impuestos por varios LLMs. Pruebe y asegúrese de que su solicitud se mantenga dentro de los límites disponibles impuestos por el proveedor del modelo.

Para ayudar a controlar el tamaño de las indicaciones, intente lo siguiente:

1. Familiarícese con los límites impuestos por el modelo que desee utilizar. Estos valores pueden diferir considerablemente de un modelo a otro, por lo que es importante saber cuál es el presupuesto disponible antes de empezar.
2. Elabore su solicitud inicial teniendo en cuenta ese presupuesto y considere cuánto desea ahorrar para cualquier elemento dinámico de la solicitud. Por ejemplo, las entradas del usuario, el historial de chat, los extractos de documentos, etc.
3. En la página de configuración del mensaje, establece un límite en el tamaño del historial final para limitar el número de turnos de conversación incluidos en el mensaje.
4. Establezca los límites de devolución de documentos en el asistente de configuración de Knowledge Base. Debe intentar encontrar el equilibrio adecuado entre proporcionar al LLM el contexto suficiente para realizar la tarea, pero no tanto como para superar los límites simbólicos o afectar negativamente a la latencia.

5. Deja un poco de margen. No haga un presupuesto para el caso típico, piense en los casos extremos y experimente con ellos, como las consultas de entrada largas, los extractos de documentos de gran tamaño o las conversaciones largas.

Pasos para construir el servidor MCP (Docker Image)

Para usar servidores MCP (Model Context Protocol) con Generative AI Application Builder en AWS, como primer paso, necesita una imagen de Docker creada y almacenada en un repositorio privado de Amazon ECR.

> **Note**
>
> Por el momento, los servidores MCP implementados actualmente en Amazon Bedrock AgentCore Runtime no se pueden exportar a GAAB. Para que los servidores MCP se conecten a los agentes creados mediante la GAAB, es necesario crearlos mediante la GAAB.

Paso 1: Cree su servidor MCP

En primer lugar, debe tener lista la implementación de su servidor MCP. Para obtener instrucciones detalladas sobre la creación de un servidor MCP, consulte la Guía para AgentCore desarrolladores de Amazon Bedrock: Creación de un servidor MCP.

Recomendamos la siguiente estructura de proyecto:

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

Para la estructura de Dockerfile, recomendamos utilizar un formato similar al siguiente ejemplo:

```
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"]
```

Paso 2: Pruebe su servidor MCP localmente

Antes de implementarlo en AWS, es importante probar el servidor MCP localmente para asegurarse de que funciona como se espera. Para obtener instrucciones detalladas sobre las pruebas locales, consulte la Guía para AgentCore desarrolladores de Amazon Bedrock: pruebe su servidor MCP localmente.

Paso 3: Implementación en Amazon ECR

Una vez que haya creado y probado su servidor MCP localmente, siga estos pasos para implementarlo en Amazon ECR:

1. Asegúrese de tener instalada la versión más reciente de AWS CLI y Docker. Para obtener más información, consulte Introducción a Amazon ECR.
2. Obtenga un token de autenticación y autentique su cliente Docker en su registro. Utilice la AWS CLI:

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

3. Cree su imagen de Docker con el siguiente comando. Para obtener información sobre cómo crear un archivo de Docker desde cero, consulta la documentación de Docker. Puedes saltarte este paso si la imagen ya está creada:

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

4. Cuando se complete la compilación, etiquete la imagen para poder enviarla a este repositorio:

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

5. Ejecute el siguiente comando para enviar esta imagen al repositorio de AWS recién creado:

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

Para obtener instrucciones de implementación completas, consulte la Guía para AgentCore desarrolladores de Amazon Bedrock: Implemente su servidor MCP en AWS.

Paso 4: Utilice el URI de ECR en la GAAB

Tras enviar correctamente la imagen de Docker a Amazon ECR, copie el URI de la imagen de la consola de ECR. Utilizará este URI al implementar su servidor MCP mediante el asistente de implementación Generative AI Application Builder en AWS.

Pasos para crear diferentes objetivos de MCP Gateway

Amazon Bedrock AgentCore Gateway le permite transformar los servicios de AWS existentes y APIs convertirlos en herramientas de MCP que pueden utilizar sus agentes. El Gateway admite varios tipos de objetivos, lo que le permite integrar varios servicios de backend sin problemas.

Se admiten los siguientes tipos de objetivos:

- Objetivos de Lambda: transforme las funciones de AWS Lambda en herramientas de MCP. Para obtener instrucciones detalladas, consulte la Guía para AgentCore desarrolladores de Amazon Bedrock: Añadir objetivos Lambda.
- Objetivos de OpenAPI: utilice las especificaciones de OpenAPI para definir y exponer APIs REST como herramientas de MCP. Para obtener instrucciones detalladas, consulte la Guía para AgentCore desarrolladores de Amazon Bedrock: esquema OpenAPI.
- Objetivos de Smithy: Cree herramientas de MCP utilizando las definiciones del modelo de Smithy para lograr integraciones de API con tipos seguros. Para obtener instrucciones detalladas, consulte la Guía para AgentCore desarrolladores de Amazon Bedrock: objetivos de Building Smithy.
- Destinos del servidor MCP: Conéctese directamente a servidores MCP externos a través de puntos finales URL, lo que le permitirá integrar los servidores MCP existentes. Para obtener instrucciones detalladas, consulte la Guía para AgentCore desarrolladores de Amazon Bedrock: objetivos de servidores MCP.

Para ver ejemplos y tutoriales adicionales sobre la creación de objetivos de MCP Gateway, visite el repositorio de AgentCore muestras de Amazon Bedrock.

Configuración de una base de conocimientos

En esta sección se describe cómo incorporar datos a la base de conocimientos que ha seleccionado para la solución. Actualmente, la solución es compatible con las bases de conocimiento de Amazon Kendra y Amazon Bedrock como bases de conocimiento para la implementación de casos de uso basados en RAG.

Amazon Kendra

Si utiliza Amazon Kendra como base de conocimientos, consulte la Guía para desarrolladores de Amazon Kendra para obtener información sobre cómo utilizar varios conectores de fuentes de datos para ayudarle a ingerir datos de una amplia gama de fuentes.

Importante: Para evitar la pérdida accidental de datos, la solución no elimina automáticamente el índice de Kendra (ya sea que lo haya creado la solución o no) cuando se elimina una implementación o una pila. Si desea eliminar su base de conocimientos y dejar de incurrir en costes, consulte la sección sobre desinstalación manual para obtener más información sobre los recursos que se conservan y cómo limpiarlos.

Bases de conocimiento de Amazon Bedrock

Las bases de conocimiento de Amazon Bedrock pueden estar respaldadas por una variedad de almacenes de vectores diferentes, cada uno con la capacidad de indexar sus datos. Para configurar y completar su base de conocimientos, consulte la Guía del usuario de Amazon Bedrock. En concreto, querrá:

- Primero configure su fuente de datos
- A continuación, configure un índice vectorial para su base de conocimientos en un almacén vectorial compatible. Tenga en cuenta que esto se puede omitir si utiliza la opción «Crear rápidamente un nuevo almacén de vectores» en la consola de Bedrock durante la creación de la base de conocimientos.
- Por último, puede crear la base de conocimientos y sincronizar las fuentes de datos configuradas.

Configuración avanzada de la base de conocimientos

Los ajustes avanzados de la base de conocimientos, como el filtrado de la base de conocimientos y el RAG con control de acceso basado en roles, están disponibles para su uso con la solución. El filtrado de la base de conocimiento se puede aplicar a cualquiera de las bases de conocimiento, mientras que el RAG con control de acceso basado en roles está disponible específicamente para Amazon Kendra.

Filtrado de bases de conocimientos

La solución le permite especificar los filtros de atributos de Amazon Kendra o los filtros de recuperación de la base de conocimiento de Bedrock al implementar un caso de uso en la sección de configuraciones avanzadas de RAG del paso de la base de conocimiento del asistente. Estos filtros definen cómo se consultan las fuentes de datos de la base de conocimientos, como las estrategias de búsqueda, los idiomas del documento subyacente al que se consulta, etc.

En ambos casos, se utiliza un objeto JSON para especificar la configuración del filtro según el formato especificado en la documentación de cada servicio (como se indica en el enlace anterior).

Ejemplo 1: Kendra AttributeFilter

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

```
 }
 }
}
```

Ejemplo 2: Bedrock RetrievalFilter

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

RAG con control de acceso basado en roles con Amazon Kendra

El control de acceso basado en roles (RBAC) permite controlar qué usuarios o grupos pueden acceder a determinados documentos del índice de Amazon Kendra o ver determinados documentos en sus resultados de búsqueda. Para configurar el RBAC para su ID de índice de Amazon Kendra con su caso de uso de Generative AI Application Builder on AWS (GAAB), siga estos pasos:

1. Configurar el índice Amazon Kendra

1. Asegúrese de haber creado un índice de Amazon Kendra y de haberle agregado al menos una fuente de datos.
2. Configure el control de acceso para su fuente de datos en función de los grupos de usuarios. Para una fuente de datos de S3, siga las instrucciones disponibles en la documentación para configurar las listas de control de acceso (ACLs) con los mismos nombres de grupo creados en su grupo de usuarios de Amazon Cognito. Esto garantiza que los usuarios solo puedan acceder a los documentos y resultados de búsqueda que estén autorizados a ver en función de su pertenencia a un grupo.

> **Note**
>
> En Control de acceso de usuarios en el índice de Kendra que ha creado, deje el control de acceso de usuarios basado en token como No. Al habilitar el control de acceso basado en roles en el paso 2, Generative AI Application Builder de AWS extrae las notificaciones correspondientes del token de autenticación del usuario y crea un filtro de atributos.

2. Implemente el caso de uso de RAG mediante el asistente de implementación de GAAB

1. Siga las instrucciones que aparecen en pantalla del asistente de implementación de la GAAB hasta llegar al paso 4 del asistente para configurar el RAG.
2. En el paso Seleccionar base de conocimientos del asistente de implementación, elija Amazon Kendra como tipo de base de conocimientos.
3. Especifique si tiene un índice de Amazon Kendra existente o si desea crear uno nuevo. Si ya tiene un índice, proporcione el ID del índice de Amazon Kendra que se ha configurado con listas de control de acceso (ACLs) basadas en grupos de usuarios.
4. Habilite la opción de control de acceso basado en roles. Esta opción garantiza que los resultados de búsqueda devueltos por el índice de Amazon Kendra se filtren en función del rol del usuario y de los permisos de grupo.
5. Revise e implemente el caso de uso.

3. Configuración de Amazon Cognito

1. Localice el grupo de usuarios de Amazon Cognito que utiliza su implementación de GAAB. Este grupo de usuarios de Amazon Cognito suele crearse mediante el conjunto de paneles CloudFormation de implementación principal.
2. Cree nuevos usuarios en el grupo de usuarios de Amazon Cognito. Al crear usuarios, seleccione la opción «Enviar una invitación por correo electrónico» para que los usuarios reciban las credenciales de inicio de sesión temporales por correo electrónico. Esto permite a los nuevos usuarios registrarse y acceder a la aplicación GAAB.
3. Cree grupos de usuarios en el grupo de usuarios de Amazon Cognito. Asegúrese de que los nombres de los grupos coincidan exactamente con los grupos configurados en su índice de Amazon Kendra. ACLs Esto es crucial para habilitar el RBAC, ya que la pertenencia al grupo del usuario determinará los resultados de búsqueda a los que puede acceder.
4. Asigne a los usuarios a los grupos adecuados en función de sus funciones y permisos de acceso. Los usuarios deben agregarse tanto al grupo requerido para la ACL del índice de Amazon Kendra como al grupo específico del caso de uso creado durante la implementación de la GAAB. Esto garantiza que los usuarios dispongan de los permisos necesarios para acceder al caso de uso específico y a los resultados de búsqueda pertinentes.

Si sigue estos pasos, habrá configurado el control de acceso basado en roles (RBAC) para su implementación de GAAB, de modo que los usuarios solo puedan acceder e interactuar con la

información y las funciones para las que están autorizados, en función del grupo de usuarios y los permisos que tengan asignados.

> **Note**
>
> Por el momento, solo Amazon Kendra admite RBAC para las bases de conocimiento del Generative AI Application Builder de AWS. Para Amazon Bedrock Knowledge Base, no se admite el RBAC, pero puede utilizar filtros de metadatos para lograr cierto nivel de filtrado. Para obtener más información, consulte la Guía del usuario de Amazon Bedrock.

Configuración de sus indicaciones

El asistente del panel de implementación incluye un paso de configuración rápida que le permite personalizar la experiencia y la plantilla que guiarán las interacciones entre los usuarios y el modelo de IA. La configuración adecuada de estos ajustes es crucial para obtener respuestas precisas y relevantes del asistente de IA.

Esta sección controla la experiencia y el comportamiento generales del mensaje de la IA.

- Longitud máxima de la plantilla de mensaje: esta configuración determina la longitud máxima (en caracteres) de la plantilla de mensaje. Un valor más alto permite proporcionar más contexto al modelo de IA, lo que podría generar respuestas más precisas. Sin embargo, las indicaciones excesivamente largas también pueden generar ruido y afectar negativamente al rendimiento. Para los modelos Amazon Bedrock, los valores predeterminados de la longitud máxima de la plantilla de solicitud (en caracteres) se calculan utilizando los límites de token del modelo subyacente. Si edita y cambia el nombre de un modelo en Bedrock, aparece resaltado el botón «Restablecer los valores predeterminados», que puede utilizarse para adoptar los valores predeterminados del modelo recién seleccionado. Para los modelos de Amazon SageMaker AI, se proporcionan valores predeterminados razonables, pero se recomienda comprobar el modelo subyacente y elegir la longitud máxima de la plantilla de solicitud e introducir las longitudes de texto en consecuencia. Consulte la sección Consejos para gestionar los límites de los tokens de los modelos para obtener más información.
- Longitud máxima del texto de entrada: esta configuración limita la longitud máxima (en caracteres) del texto introducido por el usuario. Las entradas más largas pueden contener información irrelevante, lo que aumenta el riesgo de obtener respuestas irrelevantes o inexactas del modelo de IA.

- Edición de mensajes de usuario: esta opción permite activar o desactivar la posibilidad de que los usuarios modifiquen la plantilla de mensajes a través de la interfaz de usuario del chat. La desactivación de esta función puede ayudar a mantener la coherencia y evitar cambios no deseados en el mensaje.

Plantilla de solicitud

Esta sección le permite definir la plantilla de solicitud real que utilizará el modelo de IA. La plantilla de mensajes suele seguir una estructura que incluye marcadores de posición para varios componentes, como la entrada del usuario, los pasajes de referencia y el historial de chat.

- Plantilla de mensaje: es el área de texto principal donde puedes escribir o pegar la plantilla de mensaje que desees. La plantilla debe diseñarse para proporcionar el contexto y las instrucciones necesarios para el modelo de IA. Por lo general, incluye los siguientes marcadores de posición:
  - `{input}`: Este marcador de posición es obligatorio para las implementaciones de Sagemaker AI y se sustituirá por la entrada o consulta del usuario.
  - `{history}`: Este marcador de posición es obligatorio para las implementaciones de Sagemaker AI y se sustituirá por el historial de chat de la conversación actual.
  - `{context}`: Este marcador de posición es obligatorio para las implementaciones de RAG y se sustituirá por los extractos del documento obtenidos de la base de conocimientos configurada.
- ¿Reformular la pregunta? : Esta opción (disponible solo para las implementaciones de RAG) determina si la consulta de entrada original del usuario debe reformularse o desambiguarse antes de pasarla al modelo de IA. Reformular la consulta a veces puede ayudar al modelo a comprender mejor la intención del usuario, lo que podría generar respuestas más precisas.

Al configurar la plantilla y la experiencia del mensaje, es fundamental lograr un equilibrio entre proporcionar suficiente contexto e instrucciones al modelo de IA y, al mismo tiempo, evitar información excesivamente larga o irrelevante que pueda provocar ruido o problemas de rendimiento.

Configuración avanzada de los mensajes

Esta sección le permite controlar cómo se presenta el historial de conversaciones en el modelo de IA.

- Tamaño del historial final: esta configuración determina el número de mensajes anteriores que se deben incluir en el mensaje final. Si se establece este valor en cero, no se incorporará ningún historial ni en la plantilla de mensaje ni en la plantilla de mensaje de desambiguación. Tenga en

cuenta que, incluso si se establece en cero, es necesario que exista un marcador de posición de {historial} en las plantillas de mensajes. En tiempo de ejecución, se reemplazará por una cadena vacía.

  - Nota: Se recomienda proporcionar un número par para este valor. Si se proporciona un número impar, solo se devolverá la respuesta de la IA de una interacción emparejada.
- Prefijo humano: es el prefijo que se utiliza para identificar los mensajes enviados por el usuario en el historial de conversaciones.
- Prefijo de IA: es el prefijo que se utiliza para identificar los mensajes devueltos por el modelo de IA en el historial de conversaciones.

Configuración del aviso de desambiguación

Esta sección le permite configurar el comportamiento y la plantilla para eliminar la ambigüedad de las entradas de los usuarios antes de enviarlas a la base de conocimientos configurada.

- Habilitar la desambiguación: esta opción determina si las entradas del usuario deben desambiguarse antes de enviarlas a la base de conocimientos.
- Plantilla de mensaje de desambiguación: esta es la plantilla de mensaje que se utiliza para eliminar la ambigüedad de las entradas de los usuarios cuando se conecta a una base de conocimientos. El resultado generado a partir de este mensaje se utilizará como consulta enviada a la base de conocimientos. Al deshabilitar la desambiguación, la consulta sin procesar del usuario se enviaría a la base de conocimientos sin cambios.

Por ejemplo, con la desambiguación habilitada, una consulta de seguimiento del usuario sobre «¿Cuánto cuesta?» podría desambiguarse y convertirse en «¿Cuánto cuesta renovar mi matrícula?» , lo que permite una mejor consulta de búsqueda.

Utilizando el caso de uso de Text implementado

La interfaz de usuario integrada para el caso de uso de Text está diseñada para permitir a los usuarios empresariales explorar y experimentar rápidamente con la implementación creada por el usuario administrador. Los cambios de configuración realizados por el usuario empresarial solo se aplican a su sesión. El usuario empresarial debe compartir estos cambios con el usuario administrador, quien puede actualizar la implementación base con esos cambios para que todos los puedan utilizar.

La interfaz de usuario del chat consta de los siguientes componentes:

- Ventana de chat
- Cuadro de entrada de chat
- Configuración
- Conversación clara

Ventana de chat

Mantiene diferentes turnos de la conversación. Los mensajes que comienzan por la derecha provienen del usuario empresarial y los que comienzan por la izquierda provienen del LLM configurado. En todas las respuestas del LLM aparece un pequeño icono de portapapeles que permite copiarlas fácilmente.

Cuadro de entrada de chat

En la parte inferior de la ventana de chat se encuentra el cuadro de entrada del chat. Aquí es donde los usuarios empresariales pueden introducir sus mensajes para enviarlos al LLM. Justo encima del cuadro de entrada está el estado de la conexión. Si se pierde la conexión (por ejemplo, por inactividad), se creará automáticamente una nueva conexión la próxima vez que se envíe un mensaje de chat. Se espera que esta solicitud tarde un poco más debido al tiempo de WebSocket conexión adicional.

Según la configuración específica, es posible que se aplique una longitud máxima a la entrada. Si se supera este límite, los usuarios reciben una alerta y el mensaje no se envía.

Nota: Si usa RAG con Amazon Kendra, la API Retrieve truncará las consultas a 30 palabras simbólicas. Si espera que los usuarios introduzcan más tiempo, evalúe cómo esto podría afectar al rendimiento de la búsqueda.

Configuración

Para que los usuarios empresariales puedan experimentar rápidamente con diferentes configuraciones, hay disponible un panel de ajustes que permite on-the-fly editar determinadas opciones de configuración de despliegue

(ejemplo, plantilla de solicitud). Estos cambios solo se pueden realizar al comienzo de una nueva sesión. Una vez iniciada una conversación, si se borra la conversación, se vuelven a permitir la edición de los ajustes de configuración.

Nota: Los usuarios administradores pueden optar por bloquear la configuración de una implementación. Pueden impedir las ediciones en tiempo real en el momento de la implementación mediante el asistente durante el siguiente paso.

Conversación clara

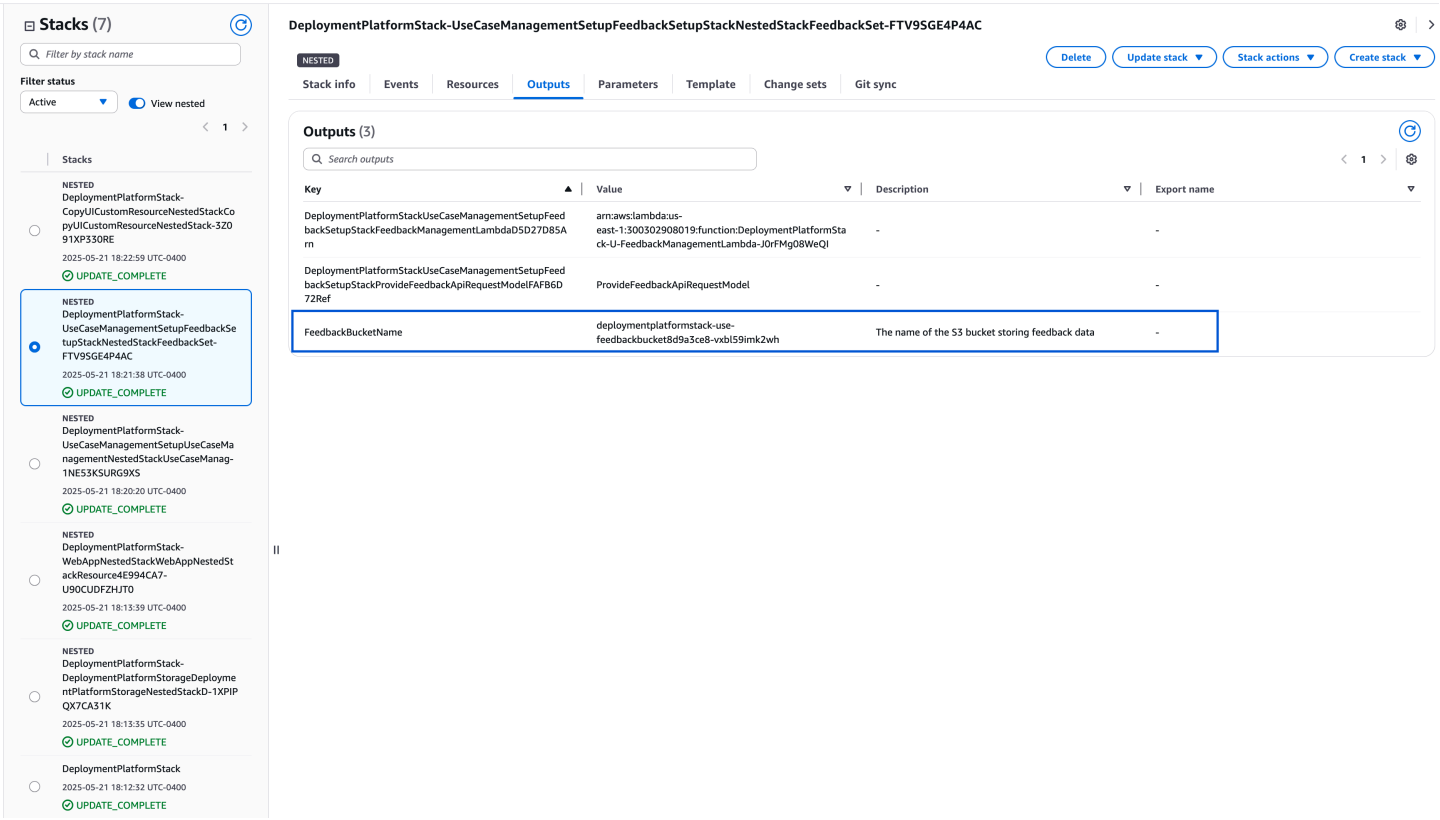
A lo largo de la conversación, la solución mantiene un historial de chat, lo que permite una experiencia conversacional. Esto permite la desambiguación de las consultas y las preguntas de seguimiento. Para restablecer una conversación y eliminar todo el historial de chat de esta interacción, selecciona *Borrar conversación *en la parte superior de la ventana de chat. Una vez que se borra la conversación, se crea una nueva sesión que permite volver a editar los ajustes.

Acceder a los comentarios recopilados por los usuarios y analizarlos

A partir de la versión 3.0.0, el panel de implementación implementa una pila de comentarios anidada que permite que los casos de uso de Text y Bedrock Agent implementados con el panel tengan la funcionalidad de recopilar comentarios para las respuestas que se generan. LLM/Agent En particular, los usuarios pueden enviar comentarios positivos o negativos junto con un comentario opcional. Si el usuario proporciona un comentario negativo, puede seleccionar una de estas categorías negativas: «Inexacto», «Incompleto o insuficiente», «Perjudicial» y «Otros». and/or

Una vez que el usuario proporciona los comentarios, estos se almacenan en un bucket de S3 dividido por ID de caso de uso, año y mes. El identificador del caso de uso se encuentra en el panel de control de implementación y el grupo de comentarios sobre S3 se encuentra en los resultados de la pila anidada de comentarios del panel de implementación:

Representa la pila de despliegues: buscar el nombre del depósito de comentarios



Los comentarios de los usuarios se envían como una solicitud de API que contiene un conjunto mínimo de información:

```
{
  "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."
}
```

A continuación, una lambda procesa esta carga útil mediante la `useCaseRecordKey` cual se identifica la configuración correcta de un caso de uso en el momento de la implementación. Esta configuración se utiliza para obtener detalles específicos de los comentarios, como el ConversationTable nombre (contiene todas las conversaciones y las secuencias de mensajes humanos y de la IA), que luego se utiliza para recuperar el y real. `userInput` `llmResponse` También se adjuntan detalles adicionales a este registro de comentarios, como el `agentId` caso de uso de Bedrock Agent y, etc. `modelProvider` `bedrockModelId`, para un caso de uso de texto que utilice esta configuración. `agentAliasId` Para obtener más información sobre cómo acceder a esta configuración, consulte la sección de mapeos de comentarios personalizados que aparece a continuación. Cada solicitud de comentarios entrante se almacena como un objeto JSON y un ejemplo de registro de comentarios puede tener el siguiente aspecto para un caso de uso de texto:

```
{
   "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 así para un caso de uso de 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"
}
```


Luego, esta retroalimentación se puede utilizar para seguir procesando, analizando y modelando los bucles de reentrenamiento o retroalimentación. También puede añadir mapeos personalizados para mejorar el registro de comentarios que se almacena en la lambda de comentarios.

Mapeos de comentarios personalizados

El panel de implementación contiene una `LLMConfigTable` que se puede encontrar en las salidas de la pila del panel de implementación con la clave. `LLMConfigTableName` `LLMConfigTable`contiene las configuraciones para cada caso de uso en función de las opciones seleccionadas por el administrador al implementar el caso de uso mediante el asistente del panel de implementación. Cada configuración de caso de uso se identifica por su. `useCaseRecordKey` Este es un ejemplo de registro de configuración de casos de uso en: `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"
   }
}
```

Si la retroalimentación está habilitada para un caso de uso, esta configuración contendrá un `FeedbackParams` objeto que permitirá incluir un `CustomMappings` objeto dentro de ella y especificar los JSONPaths campos adicionales que se añadirán al registro JSON de comentarios almacenado en el depósito de comentarios de S3. Por ejemplo, para el ejemplo de configuración de casos de uso anterior, CustomMappings contiene `NumberOfDocs` y, `ScoreThreshold` JSONPaths además, en el `CustomMappings` objeto, que comienza con la `config` raíz del. JSONPath Con esta

configuración, cada registro JSON almacenado en el depósito de comentarios de S3 empezará a obtener estos dos valores adicionales, aparte de los campos que ya se han proporcionado.

Analizando los datos de los comentarios

Los datos de comentarios se almacenan en S3 como objetos JSON. Estos son algunos enfoques para hacer que estos datos de comentarios sean más accesibles y procesables:

Uso de AWS Glue y Amazon Athena

AWS Glue y Amazon Athena ofrecen una forma sin servidor de catalogar, consultar y analizar sus datos de comentarios.

AWS Glue le permite crear un rastreador de AWS Glue que inspecciona los datos de un depósito de S3, deduce su esquema y registra todos los metadatos relevantes en un catálogo. Después de eso, se pueden utilizar servicios como Amazon Athena para consultar los datos.

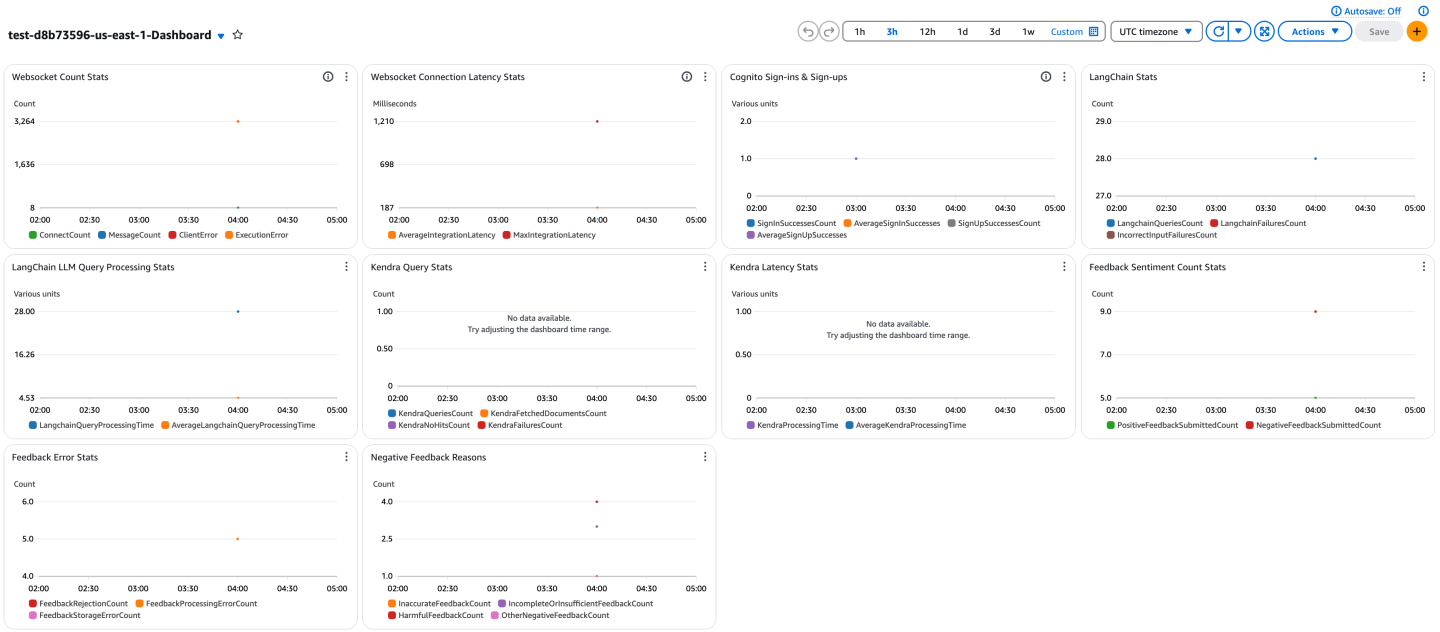
Puede consultar la documentación de AWS Athena sobre los pasos para conectar el depósito de comentarios de S3 con Amazon Athena mediante el catálogo de datos de AWS Glue. También puede utilizar algunas de las funciones más potentes de Glue para realizar tareas de extracción, transformación y carga (ETL) con estos datos y transformarlos en un formato que se adapte a sus casos de uso de análisis o reentrenamiento de modelos. Con Glue, puede realizar operaciones como filtrar los registros con determinados tipos de comentarios, rellenar la información que falte y, además, cargar estos datos en otra ubicación de almacenamiento, como otro depósito de S3 o un data store de AWS diferente.

> **Note**
>
> Según tu caso de uso, considera programar el rastreador Glue para que se ejecute periódicamente (por ejemplo, semanalmente) en lugar de cada noche para optimizar los costos, ya que los datos de comentarios pueden ser escasos.

Usa los paneles de control de la solución CloudWatch

También tiene acceso a un CloudWatch panel de control incluido con la solución, que puede proporcionarle las tendencias de las valoraciones positivas y negativas, las categorías de motivos de las valoraciones negativas, etc., según cada caso de uso. Puede encontrar este panel con el nombre de su caso de uso en los paneles de control de la consola de AWS: CloudWatch

Representa el panel de casos de uso CloudWatch



También puede crear widgets adicionales en este panel de control o crear paneles de Amazon Quick Sight.

Mejores prácticas para el análisis de los datos de comentarios

- Implemente políticas de ciclo de vida de los datos en su depósito de S3 para archivar los datos de comentarios más antiguos en niveles de almacenamiento de menor coste
- Cree un análisis independiente para cada caso de uso a fin de identificar las oportunidades de mejora específicas del modelo
- Establezca umbrales de retroalimentación que activen alertas cuando la retroalimentación negativa supere los niveles aceptables
- Exporte información crítica periódicamente para compartirla con las partes interesadas y los equipos de mejora de modelos

Visualización de las métricas de operación de una implementación

El panel de implementación y las pilas de casos de uso vienen cada uno con su propio CloudWatch panel de control que rastrea varias métricas operativas de la solución. Puede usar estos CloudWatch paneles para ayudar a comparar diferentes implementaciones. Para acceder a los paneles:

1. Vaya a la consola de CloudWatch .

2. Busque los paneles prediseñados buscando el nombre de la pila o el identificador único universal (UUID).

Por ejemplo, el caso de uso de Text incluye gráficos que muestran el número de WebSocket conexiones, el número de inicios de sesión y registro de usuarios, el tiempo que tardó el LLM en procesar una finalización, etc. Los clientes pueden usar estos gráficos para comparar varias métricas _cuantitativas de una implementación.

Example

Es difícil comparar los resultados cualitativos de varios modelos aplicados a diferentes casos de uso. Utilice la función de clonación para iniciar varias implementaciones rápidamente y poder comparar los resultados uno al lado del otro.

Acceda a la información CloudWatch de los registros

Esta solución registra los mensajes de error, advertencia, información y depuración de las funciones Lambda. Para elegir el tipo de mensajes que se van a registrar:

1. Localice la función correspondiente en la consola de AWS Lambda.
2. Agregue una variable de entorno POWERTOOLS_LOG_LEVEL.
3. Defina la variable en el tipo de mensaje aplicable.

Para obtener más instrucciones, consulte Crear variables de entorno Lambda en la Guía para desarrolladores de AWS Lambda.

En la siguiente tabla se enumeran los tipos de niveles de registro entre los que puede elegir.

Nivel	Description (Descripción)
ERROR	Los registros incluyen información sobre cualquier cosa que provoque un error en una operación.
ADVERTENCIA	Los registros incluyen información sobre cualquier elemento que pueda provocar incoherencias en la función, pero que no

Nivel	Description (Descripción)
	necesariamente provoque un error en la operación. Los registros también incluyen mensajes de ERROR.
INFO	Los registros incluyen información de alto nivel sobre el funcionamiento de la función. Los registros también incluyen mensajes de ERROR y ADVERTENCIA.
DEBUG	Los registros incluyen información que puede resultar útil a la hora de solucionar un problema con la función. Los registros también incluyen mensajes de ERROR, ADVERTENCIA e INFORMACIÓN.

Utilice el siguiente procedimiento para añadir información sobre CloudWatch los registros a esta solución.

1. Identifique los grupos de registros relevantes:
   a. Inicie sesión en la CloudFormation consola de AWS.
   b. Elija su pila de destino.
   c. Seleccione la pestaña Recursos y busque las funciones Lambda de destino.
   d. Inicie sesión en la consola de AWS Lambda y elija cada una de las funciones de Lambda de destino.
   e. Para cada una de las funciones Lambda de destino, seleccione la pestaña Supervisar y elija Ver CloudWatch registros.
   f. Copie los nombres de los grupos de registros de los que quiere extraer información.
2. Ve a la CloudWatch consola de Amazon.
3. En el menú de navegación, en Logs, selecciona Logs Insights.
4. En la página Logs Insights, selecciona la pestaña Logs.
5. Busque los nombres de los grupos de registros desde el paso 1.
6. Copie una de las siguientes consultas de ejemplo y péguela en el campo de consulta:
   a. Para identificar todas las excepciones de los clientes:

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

b. Para recuperar el recuento de invocaciones por nombre de función:

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

c. Para recuperar el recuento de invocaciones en intervalos de cinco minutos:

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

d. Para recuperar todo el rastreo de AWS X-Ray IDs:

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

e. Para recuperar los registros relacionados con un X-Ray Trace ID específico:

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

f. Para recuperar WebSocket errores no autorizados:

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

g. Para recuperar el recuento de métricas publicadas:

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

Guía para desarrolladores

En esta sección se proporciona el código fuente de la solución, una guía de integración, una guía de personalización y una referencia sobre la API.

Código fuente

Visite nuestro GitHub repositorio para descargar los archivos fuente de esta solución y compartir sus personalizaciones con otras personas.

Las plantillas de Generative AI Application Builder en AWS se generan mediante el AWS Cloud Development Kit (AWS CDK). Consulte el archivo README.md para obtener información adicional.

Guía de integración

Toda la solución está diseñada para ser fácilmente ampliable. La capa de orquestación de esta solución se creó utilizando. LangChain Puede añadir a esta solución cualquier modelo de proveedor, base de conocimientos o tipo de memoria de conversación compatible con LangChain (o con un tercero que proporcione LangChain conectores para estos componentes).

Se admite la expansión LLMs

Para añadir otro proveedor de modelos, como un proveedor de LLM personalizado, debe actualizar los tres componentes siguientes de la solución:

1. Cree una nueva pila de `TextUseCase` CDK, que despliegue la aplicación de chat configurada con su proveedor de LLM personalizado:
   a. Clona el GitHub repositorio de esta solución y configura tu entorno de compilación siguiendo las instrucciones que se proporcionan en el archivo README.md.
   b. Copie (o cree uno nuevo) el `source/infrastructure/lib/bedrock-chat-stack.ts` archivo, péguelo en el mismo directorio y cámbiele el nombre a. `custom-chat-stack.ts`
   c. Cambie el nombre de la clase del archivo por una adecuada, como. `CustomLLMChat`
   d. Puedes optar por añadir un secreto de Secrets Manager a esta pila, que almacena tus credenciales para tu LLM personalizado. Puede recuperar estas credenciales durante la invocación del modelo en la capa Lambda de chat que se describe en el siguiente párrafo.

2. Cree y adjunte una capa Lambda que contenga la biblioteca de Python del proveedor de modelos que se va a añadir. En el caso de una aplicación de chat de casos de uso de Amazon Bedrock, la biblioteca de `langchain-aws` Python incluye los conectores personalizados en la parte superior del LangChain paquete para conectarse a los proveedores de modelos de AWS (Amazon Bedrock e SageMaker IA), las bases de conocimiento (bases de conocimiento de Amazon Kendra y Amazon Bedrock) y los tipos de memoria (como DynamoDB). Del mismo modo, otros proveedores de modelos tienen sus propios conectores. Esta capa le ayuda a adjuntar la biblioteca Python de este proveedor de modelos para que pueda usar estos conectores en la capa Lambda de chat, que invoca el LLM (paso 3). En esta solución, se utiliza un empaquetador de activos personalizado para crear capas Lambda, que se adjuntan mediante aspectos de CDK. Para crear una capa nueva para la biblioteca de proveedores de modelos personalizados:

   a. Navegue hasta la `LambdaAspects` clase en el `source/infrastructure/lib/utils/lambda-aspects.ts` archivo.

   b. Siga las instrucciones sobre cómo ampliar la funcionalidad de la clase de aspectos de Lambda que se proporcionan en el archivo (por ejemplo, añadir el `getOrCreateLangchainLayer` método). Para usar este nuevo método (por ejemplo,`getOrCreateCustomLLMLayer`), actualice también la `LLM_LIBRARY_LAYER_TYPES` enumeración del `source/infrastructure/lib/utils/constants.ts` archivo.

3. Amplíe la función `chat` Lambda para implementar un generador, un cliente y un controlador para el nuevo proveedor.

   `source/lambda/chat`Contiene las LangChain conexiones de las diferentes clases LLMs junto con las clases auxiliares para crearlas. LLMs Estas clases de apoyo siguen los patrones de diseño orientado a objetos y constructores para crear el LLM.

   Cada controlador (por ejemplo`bedrock_handler.py`) crea primero un cliente, comprueba el entorno en busca de las variables de entorno necesarias y, a continuación, llama a un `get_model` método para obtener la clase LangChain LLM. A continuación, se llama al método generate para invocar el LLM y obtener su respuesta. LangChain actualmente admite la funcionalidad de streaming para Amazon Bedrock, pero no para la SageMaker IA. Según la funcionalidad de transmisión o no transmisión, se llama al WebSocket controlador (`WebsocketStreamingCallbackHandler`o`WebsocketHandler`) apropiado para enviar la respuesta a la WebSocket conexión mediante el método. `post_to_connection`

   La `clients/builder` carpeta contiene las clases que ayudan a crear un LLM Builder utilizando el patrón Builder. En primer lugar, `use_case_config` se recupera a de un almacén de configuraciones de DynamoDB, que almacena los detalles sobre el tipo de base de conocimientos,

memoria de conversación y modelo que se debe construir. También contiene detalles relevantes del modelo, como los parámetros y las instrucciones del modelo. Luego, The Builder ayuda a seguir los pasos para crear una base de conocimientos, crear una memoria de conversación para mantener el contexto de la conversación en el caso de la LLM, configurar las LangChain llamadas apropiadas para los casos de transmisión y no transmisión y crear un modelo de LLM basado en las configuraciones del modelo proporcionadas. La configuración de DynamoDB se almacena en el momento de la creación del caso de uso cuando se implementa un caso de uso desde el panel de implementación (o cuando la proporcionan los usuarios en las implementaciones de pilas de casos de uso independientes sin el panel de implementación).

La `clients/factories` subcarpeta ayuda a configurar la memoria de conversación y la clase de base de conocimientos adecuadas, en función de la configuración de LLM. Esto permite ampliarla fácilmente a cualquier otra base de conocimientos o tipos de memoria que desee que admita su implementación.

La `shared` subcarpeta contiene implementaciones específicas de la base de conocimientos y la memoria de conversación, que el creador crea instancias en las fábricas. También contiene los recuperadores de la base de conocimientos de Amazon Kendra y Amazon Bedrock a los que se recurre LangChain para recuperar documentos para los casos de uso de RAG, junto con devoluciones de llamada, que utiliza el modelo LLM. LangChain

LangChain Las implementaciones utilizan el Lenguaje de LangChain Expresión (LCEL) para componer cadenas de conversación juntas. `RunnableWithMessageHistory`La clase se utiliza para mantener el historial de conversaciones con cadenas LCEL personalizadas, lo que permite utilizar funciones como la devolución de los documentos fuente y el uso de la pregunta reformulada (o desambiguada) enviada a la base de conocimientos para enviarla también al LLM.

Para crear tu propia implementación de un proveedor personalizado, puedes:

a. Copie el `bedrock_handler.py` archivo y cree su controlador personalizado (por ejemplo,`custom_handler.py`), que creará su cliente personalizado (por ejemplo,`CustomProviderClient`) (especificado en el siguiente paso).
b. Copia `bedrock_client.py` en la carpeta de clientes. Cámbiele el nombre a `custom_provider_client.py` (o al nombre específico de su proveedor de modelos, por ejemplo`CustomProvider`). Asigne un nombre apropiado a la clase que contiene, por ejemplo, la `CustomProviderClient` que heredaLLMChatClient.

   Puede usar los métodos proporcionados `LLMChatClient` o escribir sus propias implementaciones para anularlos.

El `get_model` método crea un `CustomProviderBuilder` (consulte el paso siguiente) y llama al `construct_chat_model` método que construye el modelo de chat siguiendo los pasos del generador. Este método actúa como director en el patrón del generador.

c. Cópielo `clients/builders/bedrock_builder.py` y cámbiele el nombre `custom_provider_builder.py` y cámbiele el nombre a la clase `CustomProviderBuilder` que contiene y hereda LLMBuilder ()`llm_builder.py`. Puedes usar los métodos proporcionados por LLMBuilder o escribir tus propias implementaciones para anularlos. Los pasos del generador se invocan en secuencia dentro del `construct_chat_model` método del cliente, por ejemplo `set_model_defaultsset_knowledge_base`, y. `set_conversation_memory`

   El `set_llm_model` método crearía el modelo LLM real utilizando todos los valores establecidos mediante los métodos invocados anteriormente. En concreto, puede crear un LLM RAG (`CustomProviderRetrievalLLM`) o no RAG (`CustomProviderLLM`), en función de lo `rag_enabled variable` que se recupere de la configuración de LLM en DynamoDB.

   Esta configuración se obtiene en el método de la clase. `retrieve_use_case_config` `LLMChatClient`

d. Implemente su `CustomProviderRetrievalLLM` implementación `CustomProviderLLM` o implementación en la `llm_models` subcarpeta en función de si necesita un caso de uso de RAG o de otro tipo. La mayoría de las funcionalidades para implementar estos modelos se proporcionan en sus `RetrievalLLM` clases `BaseLangChainModel` y, respectivamente, para casos de uso que no son RAG y RAG.

   Puede copiar el `llm_models/bedrock.py` archivo y realizar los cambios necesarios para llamar al LangChain modelo que se refiera a su proveedor personalizado. Por ejemplo, Amazon Bedrock usa una `ChatBedrock` clase para crear un modelo de chat usando LangChain.

   El método generate genera la respuesta LLM mediante las cadenas LangChain LCEL.

   También puede utilizar el `get_clean_model_params` método para sanear los parámetros del modelo según LangChain los requisitos de su modelo.

Ampliación de las herramientas de Strands compatibles

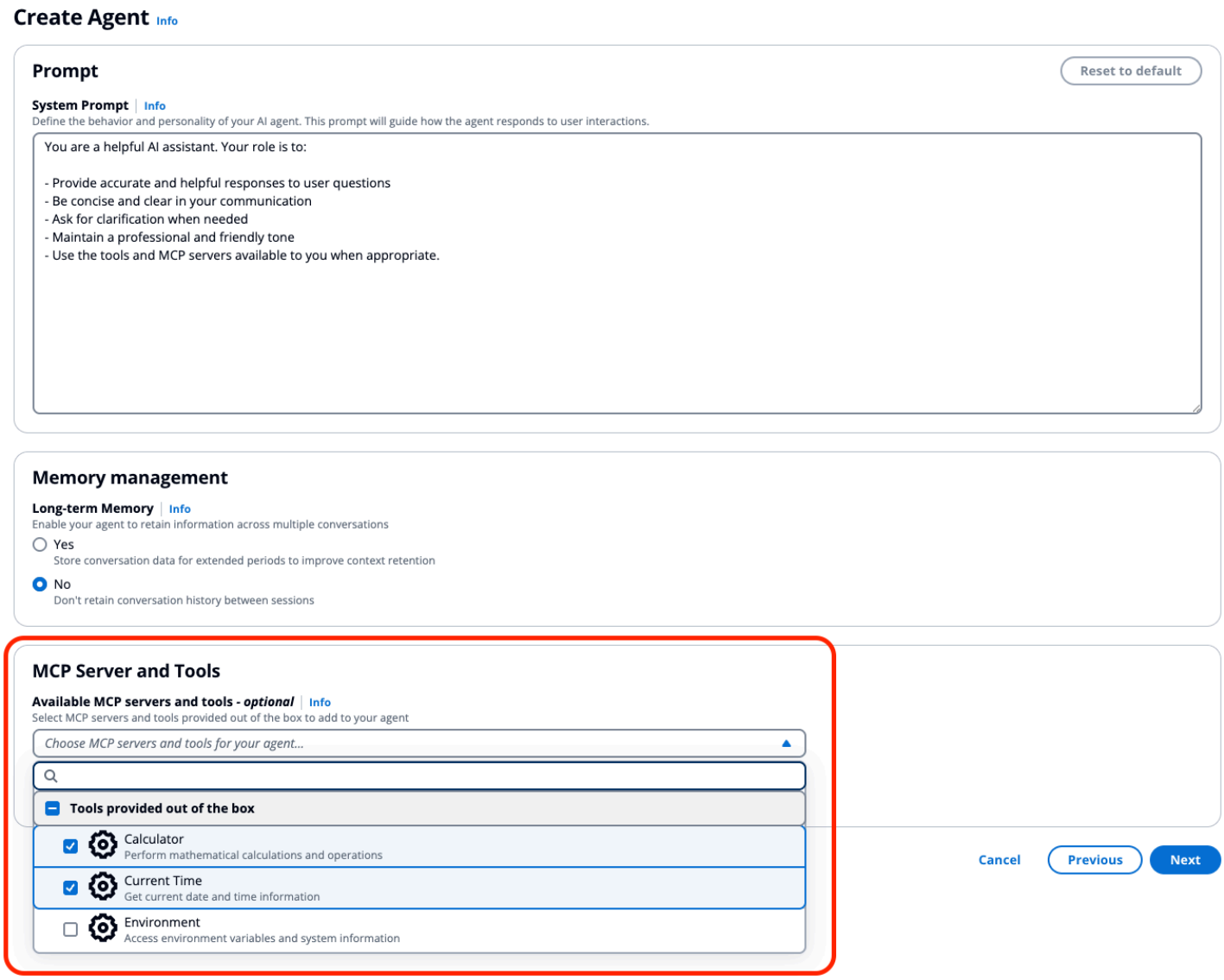
La solución le permite crear e implementar servidores MCP, agentes de IA y flujos de trabajo con múltiples agentes. Gracias a la experiencia de Agent Builder, puede conectar servidores MCP para

ofrecer a sus agentes funciones adicionales. Además de los servidores MCP, puede aprovechar las herramientas integradas que proporciona Strands (el marco subyacente utilizado por la solución).

De fábrica, la solución viene preconfigurada con las siguientes herramientas de Strands:

- Hora actual (habilitada de forma predeterminada)
- Calculadora (habilitada de forma predeterminada)
- Entorno

La selección de servidores y herramientas MCP en el asistente de Agent Builder muestra las herramientas integradas de Strands



Para ampliar sus agentes con herramientas adicionales de Strands, siga el proceso de cuatro pasos que se describe en esta sección.

Paso 1: Busca la herramienta Strands

Explore las herramientas de Strams disponibles para identificar la herramienta que desea utilizar. Cada herramienta tiene capacidades y requisitos de configuración específicos.

Por ejemplo, para añadir las capacidades de recuperación de Amazon Bedrock Knowledge Base, utilizaría la herramienta de recuperación.

Paso 2: actualice el parámetro SSM

Para que una herramienta esté disponible en la interfaz de usuario de implementación de Agent Builder, actualice el parámetro AWS Systems Manager Parameter Store que define qué herramientas de Strands son compatibles.

1. Diríjase al almacén de parámetros de AWS Systems Manager en su cuenta de AWS.
2. Localice el parámetro: `/gaab/<stack-name>/strands-tools`
3. Añada la configuración de la herramienta al final de la lista existente mediante la siguiente estructura JSON:

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

Campo	Description (Descripción)
name	El nombre para mostrar se muestra en la interfaz de usuario de Agent Builder
description	Breve descripción de la funcionalidad de la herramienta

Campo	Description (Descripción)
value	El nombre exacto de la herramienta tal y como se define en el paquete de herramien tas Strands
categoría	Categoría organizativa para agrupar herramientas en la interfaz de usuario
es el valor predeterminado	Si la herramienta debe estar habilitada de forma predeterminada para los agentes nuevos

Paso 3: Configurar las variables de entorno

Muchas herramientas de Strands requieren variables de entorno para la configuración. Puede configurar estas variables de dos maneras:

Opción 1: configuración directa en AgentCore tiempo de ejecución

Actualice el agente implementado directamente en Amazon Bedrock AgentCore Runtime con las variables de entorno necesarias.

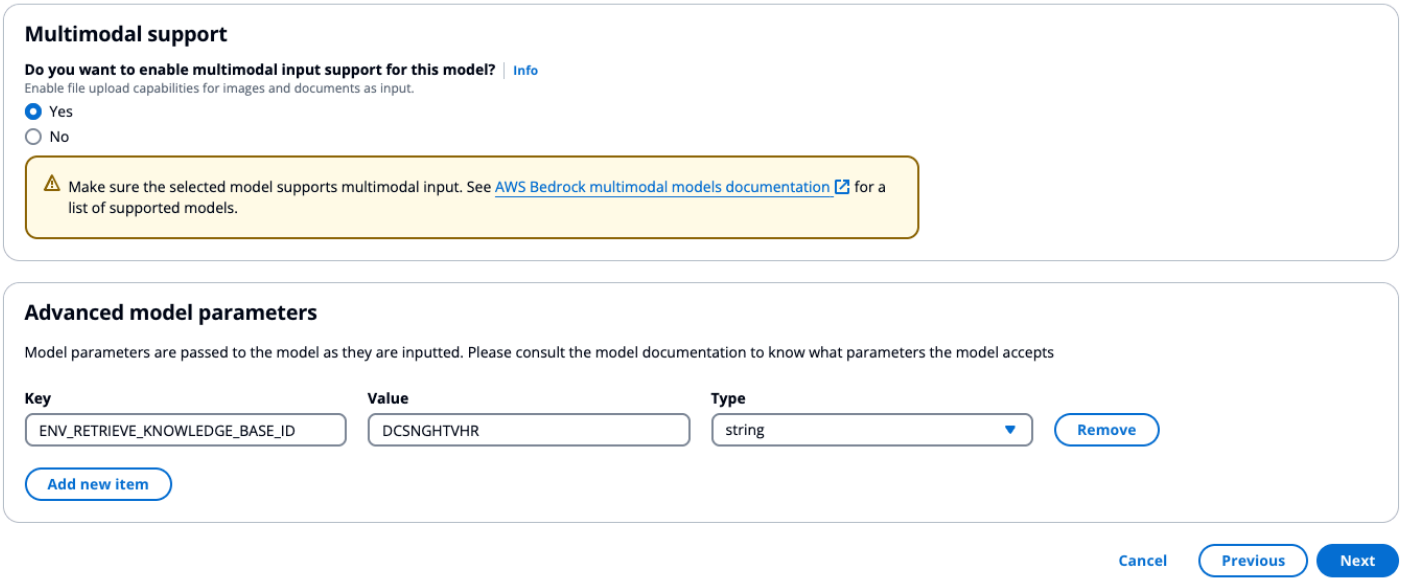
Opción 2: Modele los parámetros en el asistente de despliegue

Añada variables de entorno durante el paso de selección del modelo en el asistente de Agent Builder mediante la sección Parámetros del modelo. Las variables de entorno que siguen la convención de nomenclatura se `ENV_<ALL_CAPS_TOOL_NAME>_<env_variable_name>` cargarán automáticamente durante el tiempo de ejecución en el entorno de ejecución del agente, como`<env_variable_name>`.

Por ejemplo:

- `ENV_RETRIEVE_KNOWLEDGE_BASE_ID` se convertirá en `KNOWLEDGE_BASE_ID`
- `ENV_RETRIEVE_MIN_SCORE` se convertirá en `MIN_SCORE`

Sección de parámetros avanzados del modelo que muestra la configuración de ENV_RETRIEVE_KNOWLEDGE_BASE_ID



Consulte la documentación o el código fuente de la herramienta específica para identificar las variables de entorno requeridas. Para la herramienta de recuperación, puede encontrar las opciones de configuración en el código fuente.

Paso 4: Añadir permisos de IAM

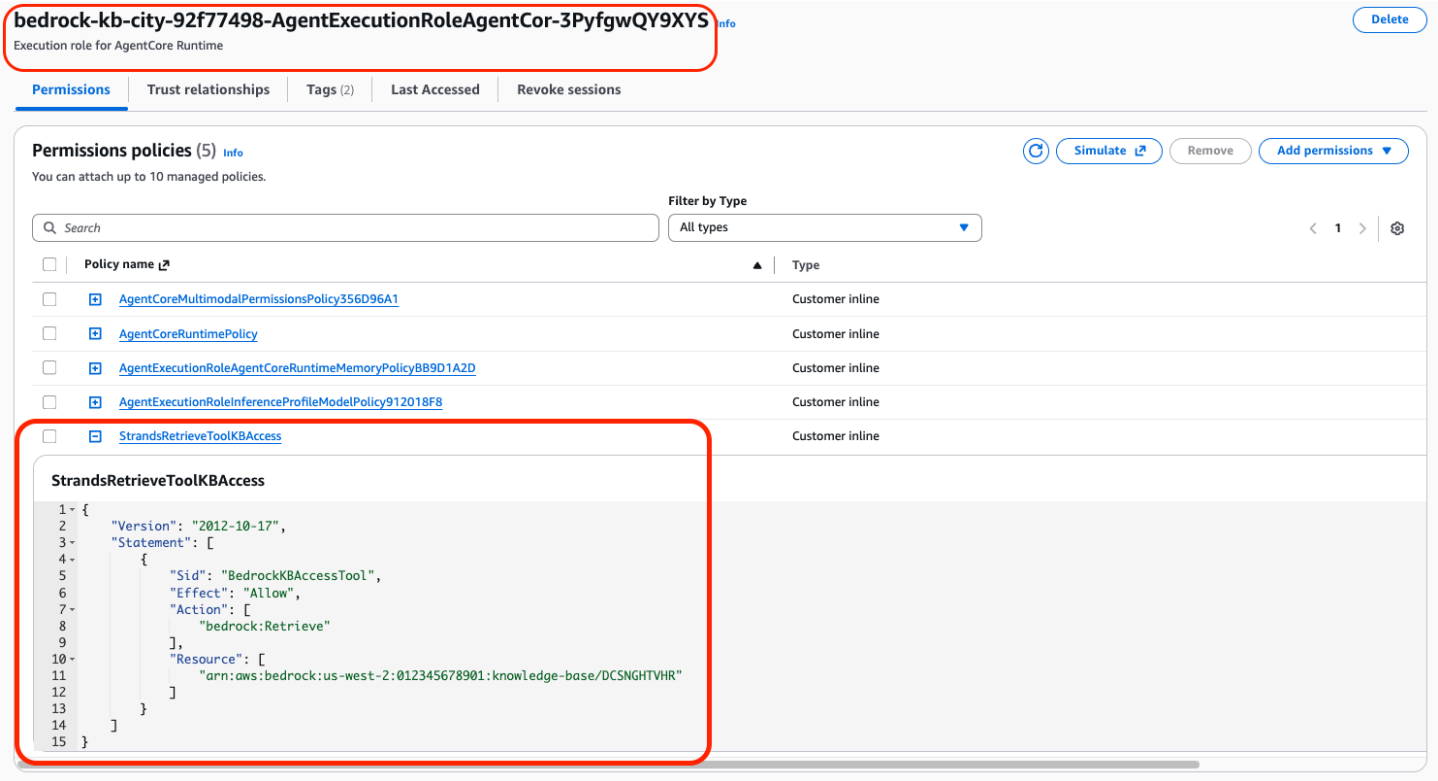
Añada manualmente los permisos de IAM necesarios a su función AgentCore de ejecución en tiempo de ejecución para que el agente pueda utilizar la herramienta.

Por ejemplo, para usar la herramienta de recuperación con las bases de conocimiento de Amazon Bedrock:

1. Navegue hasta la consola de IAM en su cuenta de AWS.
2. Localice la función AgentCore de ejecución en tiempo de ejecución de su agente.
3. Añada el siguiente permiso:

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

Consola de IAM que muestra la StrandsRetrieveTool KBAccess política asociada a la función de ejecución en AgentCore tiempo de ejecución

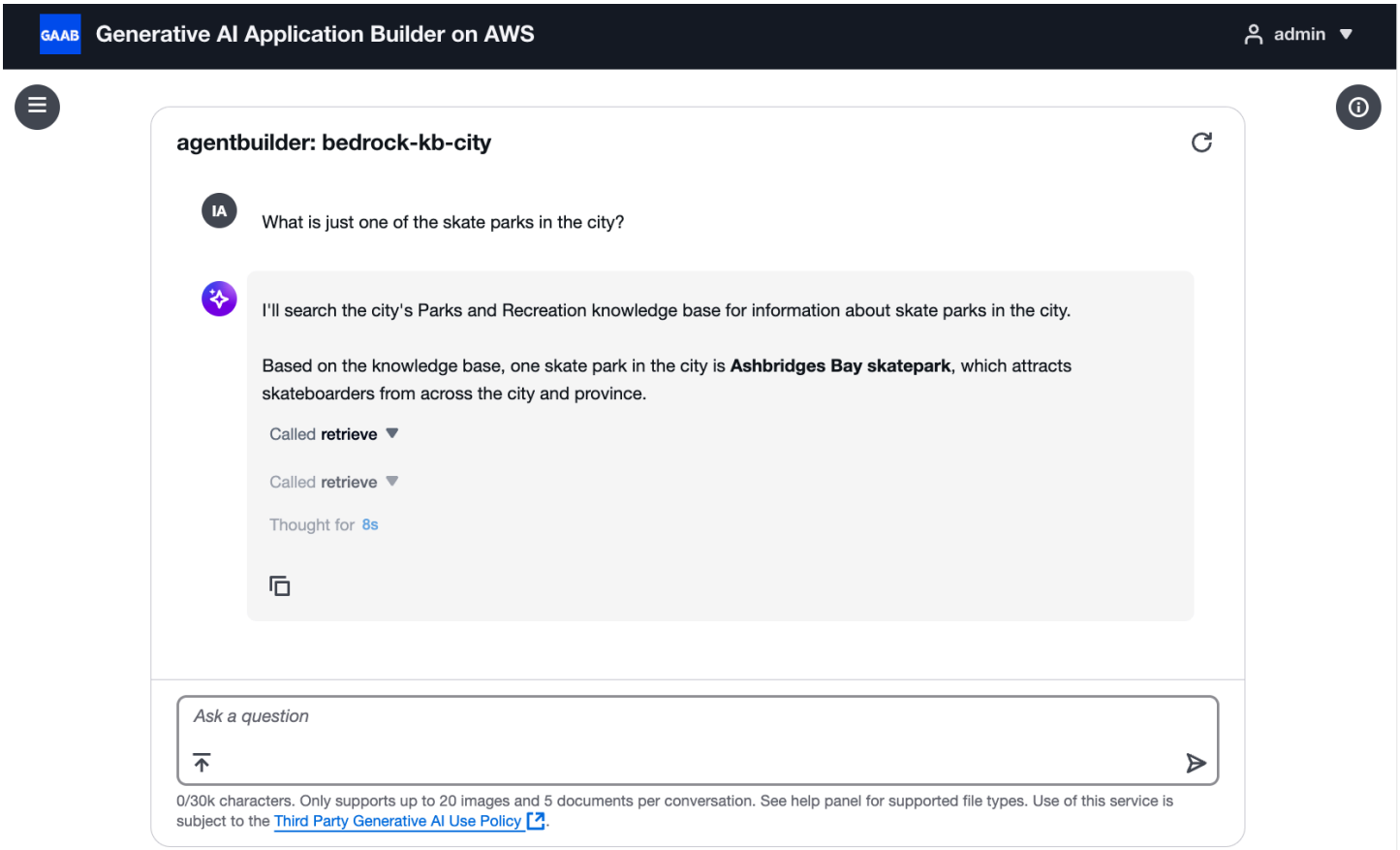


Los permisos específicos necesarios variarán en función de la herramienta. Consulte la documentación de la herramienta y la documentación del servicio de AWS para determinar los permisos de IAM adecuados.

Paso 5: Pruebe el agente

Tras completar los pasos de configuración, pruebe el agente para comprobar que la herramienta funciona correctamente. Debería ver las invocaciones a las herramientas en los registros de ejecución y las respuestas del agente.

El agente utiliza correctamente la herramienta de recuperación para responder a una pregunta sobre los parques de patinaje



> **Note**
>
> Para obtener una lista completa de las herramientas de Strands disponibles y sus capacidades, consulte la documentación de las herramientas comunitarias de Strands.

Ampliar las bases de conocimiento y los tipos de memoria de conversación compatibles

Para añadir sus implementaciones de la memoria de conversaciones o la base de conocimientos, añada las implementaciones necesarias a la `shared` carpeta y, a continuación, edite las fábricas y las enumeraciones correspondientes para crear una instancia de estas clases.

Cuando proporcione la configuración de LLM, que se almacena en el almacén de parámetros, se crearán la memoria de conversación y la base de conocimientos adecuadas para su LLM. Por ejemplo, cuando `ConversationMemoryType` se especifica como DynamoDB, se crea una instancia `DynamoDBChatMessageHistory` de (disponible en el `shared_components/memory/ddb_enhanced_message_history.py` interior). Cuando `KnowledgeBaseType` se

especifica como Amazon Kendra, se crea una instancia de `KendraKnowledgeBase` (disponible en el interior`shared_components/knowledge/kendra_knowledge_base.py`).

La creación y el despliegue del código cambian

Cree el programa con el `npm run build` comando. Una vez resueltos los errores, ejecútelo `cdk synth` para generar los archivos de plantilla y todos los activos de Lambda.

1. Puede usar el `0/stage-assets.sh` script para almacenar manualmente cualquier activo generado en el depósito de almacenamiento provisional de su cuenta.
2. Usa el siguiente comando para implementar o actualizar la plataforma:

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

   Todos CloudFormation los parámetros de AWS adicionales también deben proporcionarse junto con el AdminUserEmailparámetro.

Guía de personalización

Administrar el grupo de usuarios de Cognito

Cuando se implementa el panel de implementación, se crea un grupo de usuarios de Amazon Cognito junto con un usuario administrador para proporcionar autenticación a la aplicación. Este grupo de usuarios se comparte en el panel de implementación y en todos los casos de uso. El usuario administrador creado al implementar el panel tiene acceso automático a todos los casos de uso implementados mediante el panel. Este mecanismo se proporciona a través de grupos de grupos de usuarios de Amazon Cognito.

Cuando se implementa un caso de uso desde el panel de control, si se proporciona un correo electrónico, se creará un usuario en el grupo de usuarios compartido, junto con un grupo de usuarios con el nombre del caso de uso específico. Luego, el usuario recién creado se agrega al grupo, lo que le otorga acceso al caso de uso.

Si desea añadir un usuario adicional a un caso de uso determinado, puede hacerlo creando un usuario en el grupo de usuarios de Cognito y añadiéndolo a los grupos correspondientes a los casos de uso a los que desea que el usuario tenga acceso. Para obtener una step-by-step guía, consulte Creación de un nuevo usuario en la consola de administración de AWS.

Del mismo modo, si desea crear usuarios administradores adicionales, debe crear un usuario nuevo y añadirlo al grupo de administradores del grupo de usuarios.

Los nombres de usuario se crean tomando la parte del correo electrónico proporcionada antes del UUID del `@` caso de uso generado y añadiéndole el UUID del caso de uso generado (o, `-admin` en el caso del usuario administrador).

En la pestaña Grupos, puede ver que se han creado automáticamente un grupo de administradores y un grupo para cada caso de uso con el nombre del caso de uso (tal como se indica en el asistente) y el UUID del caso de uso.

Referencia de la API

En esta sección, se proporcionan referencias de API para la solución.

Panel de implementación

API de REST	Método HTTP	Funcionalidad	Personas que llaman autorizadas
`/deployments`	`GET`	Obtenga todos los despliegues.	Token JWT autenticado de Amazon Cognito
`/deployments`	`POST`	Crea una implementación de un nuevo caso de uso.	Token JWT autenticado de Amazon Cognito
`/deployments/{useCaseId}`	`GET`	Obtiene los detalles de la implementación de una sola implementación.	Token JWT autenticado de Amazon Cognito
`/deployments/{useCaseId}`	`PATCH`	Actualiza una implementación determinada.	Token JWT autenticado de Amazon Cognito

API de REST	Método HTTP	Funcionalidad	Personas que llaman autorizadas
`/deployments/{useCaseId}`	`DELETE`	Elimina una implementación determinada.	Token JWT autenticado de Amazon Cognito
`/model-info/use-case-types`	`GET`	Obtiene los tipos de casos de uso disponibles para la implementación	Token JWT autenticado de Amazon Cognito
`/model-info/{useCaseType}/providers`	`GET`	Obtiene los proveedores de modelos disponibles para el tipo de caso de uso dado	Token JWT autenticado de Amazon Cognito
`/model-info/{useCaseType}/{providerName}`	`GET`	Obtiene IDs los modelos disponibles para un proveedor y un tipo de caso de uso determinados	Token JWT autenticado de Amazon Cognito
`/model-info/{useCaseType}/{providerName}/{modelId}`	`GET`	Obtiene la información sobre el modelo dado, incluidos los parámetros predeterminados.	Token JWT autenticado de Amazon Cognito

> **Note**
>
> Los archivos OpenAPI y Swagger también se pueden exportar desde API Gateway para facilitar la integración con la API. Consulte Exportar una API REST desde API Gateway.

Cargas útiles POST y PATCH

Consulte a continuación un ejemplo de una carga útil POST para el `/deployments` punto final, que creará un nuevo caso de 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
 }
 }

}
```

En cuanto a las actualizaciones, la estructura es la misma que la anterior, con algunas advertencias:

- El nombre del caso de uso no se puede cambiar
- Un caso de uso solo puede cambiar los grupos de seguridad y las subredes una vez que se ha implementado en una VPC. La VPC en sí no se puede cambiar.
- Si se creó un índice de Kendra para usted como base de conocimientos, no podrá cambiar la configuración de ese índice (por ejemplo,, KendraIndexName) QueryCapacityUnits

Caso de uso compartido APIs

Los siguientes puntos finales de la API REST están disponibles para los casos de uso de Text y Bedrock Agent:

API de REST	Método HTTP	Funcionalidad	Personas que llaman autorizadas
`/details/{useCaseConfigKey}`	`GET`	Obtiene los detalles de configuración de un caso de uso específico.	Token JWT autenticado de Amazon Cognito

WebSocket API	Funcionalidad	Llamantes autorizados
`/$connect`	Inicie la WebSocket conexión y autentique al usuario.	Token JWT autenticado de Amazon Cognito
`/$disconnect`	Se llama al punto final cuando se ha WebSocket desconectado una conexión.	Token JWT autenticado de Amazon Cognito

API de detalles de casos de uso

El punto final de la API de detalles recupera información sobre un caso de uso específico:

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

Este punto final devuelve los detalles de configuración de un caso de uso específico, incluidos los parámetros del modelo, la configuración de la base de conocimientos y otra información de implementación. Se requiere un token JWT autenticado de Amazon Cognito para su autorización.

Caso de uso de texto

WebSocket API	Funcionalidad	Llamantes autorizados
`/sendMessage`	Envía el mensaje de chat del usuario al WebSocket para que lo procese con la experiencia LLM configurada.	Token JWT autenticado de Amazon Cognito

API de REST	Método HTTP	Funcionalidad	Personas que llaman autorizadas
`/feedback/{useCaseId}`	`POST`	Envía los comentarios de los usuarios para un caso de uso específico.	Token JWT autenticado de Amazon Cognito

Cargas útiles de envío de mensajes

Si te estás integrando directamente con la `/sendMessage` API, debes seguir los siguientes formatos de carga útil de solicitud y respuesta.

Solicita la carga útil

```
{
 "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
}
```

Nombre del parámetro	Tipo	Description (Descripción)
action	`String`	Actualmente, solo admitimos la acción «enviar mensaje» en WebSocket
pregunta	`String`	La entrada del usuario para enviarla al LLM
ID de conversación	`String`	Un UUID que identifica la conversación. Si no se proporciona, se creará una

Nombre del parámetro	Tipo	Description (Descripción)
		nueva conversación y se mostrará el ID de conversación en la respuesta. Todos los mensajes subsiguientes de esa conversación (en los que desees que se history/context retengan) deberán incluir el identificador de conversación en ese mensaje.
Plantilla de solicitud	`String` [Opcional]	Anula la plantilla de solicitud de este mensaje. Si está vacía o no se proporciona, se utilizará de forma predeterminada la solicitud establecida en el momento de la implementación. Debe tener los marcadores de posición adecuados especificados para la configuración dada (es decir, {historial} y {entrada} para las implementaciones de IA de Sagemaker que no sean RAG), con la adición de {contexto} si se utiliza RAG para todas las implementaciones.

Nombre del parámetro	Tipo	Description (Descripción)
AuthToken	`String` [Opcional]	AccessToken obtenido del flujo de autenticación de cognito. Esto es necesario cuando se invoca un punto final websocket de chat configurado para RAG con control de acceso basado en roles (RBAC). La lista de reclamos de cognito:groups de este token JWT se utiliza para controlar el acceso a los documentos del índice Kendra. Este parámetro no es obligatorio para los casos de uso que no sean de RAG. Tampoco es obligatorio para los casos de uso de RAG en los que el RBAC esté desactivado.

Cargas útiles de respuesta

Pregunta y respuesta

La WebSocket API responderá con 1 (si la transmisión está deshabilitada) o varios (si la transmisión está habilitada) objetos JSON estructurados de la siguiente manera para cada consulta.

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

Nombre del parámetro	Tipo	Description (Descripción)
data	`String`	Una parte de la respuesta del LLM si la transmisión está habilitada, o la respuesta completa. Si utiliza la transmisión, se enviará una respuesta de este formato con el contenido de datos END_CONVERSATION para indicar el final de la respuesta a una sola pregunta.
ID de conversación	`String`	El ID de la conversación a la que pertenece esta respuesta de SourceDocument.

Respuesta del documento fuente

Si ha configurado su caso de uso de RAG para devolver los documentos fuente, también recibirá la siguiente carga útil al final de cada respuesta para cada documento fuente utilizado para crear la respuesta.

```
{
 "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"
}
```

Nombre del parámetro	Tipo	Description (Descripción)
extracto	`String`	Un extracto del documento fuente.
location	`String`	Ubicación del documento fuente. Esto dependerá de las fuentes de datos utilizadas y del tipo de base de conocimie ntos, pero podrían ser cosas como s3 URIs o sitios web.
puntuación	`Number` | `String`	La confianza en que el documento corresponde a la pregunta planteada. Será un valor flotante de 0 a 1 para Bedrock y una cuerda (por ejemplo, ALTO, BAJO, etc.) para Kendra.
título_del documento	`String`	Título del documento fuente devuelto. Solo disponible cuando se usa Kendra.
document_id	`String`	ID del documento fuente devuelto. Solo disponible cuando se usa Kendra.
atributos_adicionales	`String`	Este campo contendrá todos los atributos adicionales del documento según los personalice en su base de conocimientos en el momento de la ingesta.
ID de conversación	`String`	El ID de la conversación a la que pertenece esta respuesta de SourceDocument.

Carga útil de la API de comentarios

A continuación, se muestra un ejemplo de una carga útil POST para el `/feedback/{useCaseId}` punto final, que enviará los comentarios de los usuarios para un caso de uso específico:

```
{
  "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 de uso de Bedrock Agent

WebSocket API	Funcionalidad	Llamantes autorizados
`/invokeAgent`	Envía el mensaje del usuario al WebSocket para que lo procese con el agente configurado.	Token JWT autenticado de Amazon Cognito

Cargas útiles de InvokeAgent

Si se va a integrar directamente con el`/invokeAgent API`, debe seguir los siguientes formatos de carga útil de solicitud y respuesta.

Carga de solicitud

```
{
"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
}
```

Nombre del parámetro	Tipo	Description (Descripción)
action	`String`	Solo apoyamos la `invokeAgent` acción en relación con. WebSocket
Texto de entrada	`String`	La entrada del usuario que se va a enviar al LLM.
ID de conversación	`String[Optional]`	Un UUID que identifica de forma exclusiva la conversación. Si no proporciona este valor, la solución crea una nueva conversación y devuelve el ID de conversación en la respuesta. Todos los mensajes subsiguientes de esa conversación (en los que desee conservar el historial y el contexto) incluyen allí el identificador de conversación.
Token de autenticación	`String[Optional]`	AccessToken obtenido del flujo de autenticación de Amazon Cognito. Este parámetro no es obligatorio. Si lo proporciona, se validará el token JWT. Esto ayuda a facilitar la ampliación de esta solución.

Cargas útiles de respuesta

Pregunta y respuesta

La WebSocket API responderá con uno (si la transmisión está deshabilitada) o varios (si la transmisión está habilitada) objetos JSON estructurados de la siguiente manera para cada consulta.

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

Nombre del parámetro	Tipo	Description (Descripción)
data	`String`	La respuesta de la invocación del agente.
ID de conversación	`String`	El ID de la conversación.

Referencia

Esta sección incluye información sobre la recopilación de datos para esta solución, sugerencias sobre los recursos relacionados y una lista de los desarrolladores que han contribuido a esta solución.

Proveedores de LLM compatibles

La solución se puede integrar con los siguientes proveedores de LLM:

1. Amazon Bedrock
   - Documentación: https://aws.amazon.com/bedrock/
   - Modelos compatibles:
     - Amazon
       - Nova Lite
       - Nova Micro
       - Nova Pro
     - AI21 Laboratorios
       - Jamba 1.5 Mini
       - Jamba 1.5 Large
     - Anthropic
       - Haiku Claude v3
       - Soneto Claude v3.5
       - Claude v3.7 Sonnet (mediante el uso de perfiles de inferencia)
     - Cohere
       - Command R
       - Command R+
     - Deepseek
       - Deepseek-R1 (mediante el uso de perfiles de inferencia)
     - Meta
       - Llama 3
       - Llama 3.2 (mediante el uso de perfiles de inferencia)

    - Mistral AI
        - Mistral 7B Instruct
        - Mistral 8x7B Instruct
    - Inferencia entre regiones
        - Posibilidad de utilizar perfiles de inferencia definidos en la misma región que el panel de implementación
2. Amazon SageMaker AI
    - Documentación: https://aws.amazon.com/sagemaker/
    - Modelos compatibles: modelos de texto a texto

Para conocer los parámetros más recientes del modelo, las mejores prácticas y los usos recomendados, consulte la documentación de los proveedores de modelos.

Recopilación de datos

Esta solución envía métricas operativas a AWS (los «datos») sobre el uso de esta solución. Utilizamos estos datos para comprender mejor cómo utilizan los clientes esta solución y los servicios y productos relacionados. La recopilación de estos datos por parte de AWS está sujeta al Aviso de privacidad de AWS.

Colaboradores

- Tarek Abdunabi
- Maid Arbash
- George Bearden
- Mukit Bin Momin
- Michael Connor
- Johnny Duval
- Nihit Kasabwala
- Ahern Knox
- Simón Krol
- Michael Lin

- Tim Mekari
- Ibrahim Mohamed
- Omar Radwan Mohsen
- James Nixon
- Dekshitha Ravikumar
- Jae Shim
- Ajay Swamy
- Mohammed Taha
- Reet Takkar
- Dimitri Tchikatilov
- Jason Wreath
- Kamyar Ziabari

Revisiones

Fecha de publicación: octubre de 2023 (última actualización: enero de 2025)

Consulte el archivo ChangeLog.md en el GitHub repositorio para ver todos los cambios y actualizaciones notables del software. El registro de cambios proporciona un historial claro de mejoras y correcciones de cada versión.

Avisos

Es responsabilidad de los clientes realizar su propia evaluación independiente de la información que contiene este documento. El presente documento: (a) tiene solo fines informativos, (b) representa las ofertas y prácticas actuales de los productos de AWS, que están sujetas a cambios sin previo aviso, y (c) no supone ningún compromiso ni garantía por parte de AWS y sus filiales, proveedores o licenciantes. Los productos o servicios de AWS se proporcionan “tal cual” sin garantías, declaraciones ni condiciones de ningún tipo, ya sean expresas o implícitas. Las responsabilidades y obligaciones de AWS con respecto a sus clientes se controlan mediante los acuerdos de AWS y este documento no forma parte ni modifica ningún acuerdo entre AWS y sus clientes.

Generative AI Application Builder en AWS se licencia según los términos de la licencia Apache versión 2.0.

Important

El generador de aplicaciones de IA generativa en AWS le permite crear e implementar aplicaciones de inteligencia artificial generativa en AWS mediante el uso del modelo de IA generativa que prefiera, incluidos los modelos de IA generativa de terceros que puede elegir usar y que AWS no posee o sobre los que no tiene ningún control («modelos de IA generativa de terceros»).

Su uso de los modelos de IA generativa de terceros se rige por las condiciones que le proporcionaron los proveedores de modelos de IA generativa de terceros cuando adquirió su licencia para usarlos (por ejemplo, sus condiciones de servicio, acuerdo de licencia, política de uso aceptable y política de privacidad).

Usted es responsable de garantizar que el uso de los modelos de IA generativa de terceros cumpla con las condiciones que los rigen y con las leyes, normas, reglamentos, políticas o normas que se le apliquen.

También eres responsable de realizar tu propia evaluación independiente de los modelos de IA generativa de terceros que utilices, incluidos sus resultados y de la forma en que los proveedores de modelos de IA generativa de terceros utilizan los datos que se les puedan transmitir en función de tu implementación. AWS no ofrece ninguna declaración o garantía con respecto a los modelos de IA generativa de terceros, que son «contenido de terceros» en virtud de su acuerdo con AWS. El generador de aplicaciones de IA generativa en AWS se ofrece como «contenido de AWS» en virtud de su acuerdo con AWS.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.