實作指南

AWS 上的生成式 AI 應用程式建置器

AWS 上的生成式 AI 應用程式建置器: 實作指南

Table of Contents

此解決方案有助於開發、快速實驗和部署生成式人工智慧 (AI) 應用程式

AWS 上的生成式 AI 應用程式建置器可促進生成式人工智慧 (AI) 應用程式的開發、快速實驗和部署，而不需要 AI 的深入經驗。此 AWS 解決方案可協助您加速開發並簡化實驗：

- 擷取您的業務特定資料和文件
- 評估和比較大型語言模型 (LLMs的效能
- 使用 AI 代理器執行多步驟任務和工作流程
- 快速建置可擴展的應用程式，並使用企業級架構部署這些應用程式

AWS 上的生成式 AI 應用程式建置器包含與下列項目的整合：

- Amazon Bedrock 上提供的 LLMs
- 您已部署在 Amazon SageMaker AI 上的 LLMs
- 適用於擷取增強生成的 Amazon Bedrock 知識庫 (RAG) https://aws.amazon.com/what-is/retrieval-augmented-generation/
- Amazon Bedrock Guardrails 可實作保護措施並減少幻覺
- Amazon Bedrock 代理程式可建置可執行任務協調和完成的代理程式工作流程
- Amazon Bedrock AgentCore 可透過延長執行時間支援來建置、部署和管理生產就緒 AI 代理器
- 適用於企業資料和工具整合的模型內容通訊協定 (MCP) 伺服器

此外，此解決方案使用 LangChain 連接器啟用與您選擇的模型的連線。這些連接器可在搭配 解決方案部署的 AWS Lambda 函數中使用。您可以從無程式碼部署精靈開始建置生成式 AI 應用程式，以進行對話式搜尋、AI 產生的聊天機器人、文字產生和文字摘要。

此實作指南提供 AWS 解決方案上的生成式 AI 應用程式建置器概觀、其參考架構和元件、規劃部署的考量事項，以及將解決方案部署到 Amazon Web Services (AWS) 雲端的組態步驟。

本指南適用於解決方案架構師、商業決策者、DevOps 工程師、資料科學家和雲端專業人員，他們想要在其環境中在 AWS 上實作生成式 AI 應用程式建置器。

使用此導覽表快速找到這些問題的答案：

如果您想要 . . .	讀取 . . .
了解執行此解決方案的成本。 執行此解決方案的預估成本會根據您部署的元件和查詢數量而有所不同。 在美國東部（維吉尼亞北部）區域中，使用預設參數和 100 名作用中使用者執行部署儀表板一個月的成本約為每月 20.12 USD。 對於使用 LLM 每天執行 100 個查詢的 1 個商業使用者，未使用 RAG 部署的文字使用案例成本約為每月 12.39 USD。 啟用 RAG 的使用案例，搭配支援每天 8，000 次互動的 Amazon Kendra 索引，其成本約為每月 204.26 USD，加上知識庫的成本。	成本
了解此解決方案的安全考量。	安全性
了解如何規劃此解決方案的配額。	配額
了解哪些 AWS 區域支援此解決方案。	支援的 AWS 區域
檢視或下載此解決方案中包含的 AWS CloudFormation 範本，以自動部署此解決方案的基礎設施資源（「堆疊」）。	AWS CloudFormation 範本
存取原始程式碼，並選擇性地使用 AWS 雲端開發套件 (AWS CDK) 來部署解決方案。	GitHub 儲存庫

功能和優勢

AWS 上的生成式 AI 應用程式建置器解決方案提供下列功能：

快速實驗

此解決方案可讓使用者透過移除部署具有不同組態的多個執行個體所需的繁重工作，並比較輸出和效能，來快速實驗。使用各種 LLMs、提示工程、企業知識庫、護欄、AI 代理器和其他參數的多種組態進行實驗。

選擇和可設定性

透過各種 LLMs 的預先建置連接器，例如透過 Amazon Bedrock 提供的模型，此解決方案可讓您靈活地部署您選擇的模型，以及您偏好的 AWS 和領導 FM 服務。您也可以啟用 Amazon Bedrock 代理程式來執行各種任務和工作流程。

客服人員建置器

使用完整的生命週期管理來建置和部署生產就緒 AI 代理器。設定系統提示、整合企業工具和資料存取的模型內容通訊協定 (MCP) 伺服器，以及啟用記憶體功能以在對話之間保留內容。代理程式部署在 Amazon Bedrock AgentCore 上，具有延長的執行時間支援和即時串流回應。

工作流程建置器

使用階層委派，將多個客服人員建置器客服人員協調到複雜的工作流程中。建立主管代理程式，以自動選取和協調專門的 Agent Builder 代理程式來處理多步驟任務。設定客服人員描述、委派策略和工作流程層級記憶體，同時重複使用現有的客服人員建置器部署。

生產就緒

此解決方案採用 AWS Well-Architected 設計原則，提供企業級安全性和可擴展性，具有高可用性和低延遲，確保與具有高效能標準的應用程式無縫整合。

可擴展的模組化架構

透過整合現有的專案或原生連接其他 AWS 服務，擴展此解決方案的功能。由於這是開放原始碼應用程式，因此您可以使用隨附的 LangChain 協同運作層或 Lambda 函數來連接您選擇的服務。

與 Service Catalog AppRegistry 和 Application Manager 整合，AWS Systems Manager 的功能

此解決方案包含 Service Catalog AppRegistry 資源，可將解決方案的 CloudFormation 範本及其基礎資源註冊為 AWS Service Catalog AppRegistry 和 AWS Systems Manager Application Manager 中的應用程式。透過此整合，您可以集中管理解決方案的資源。

客服人員建置器與 Bedrock 客服人員使用案例

此解決方案提供兩種不同的方法來使用 AI 代理器，每種都適用於不同的使用案例和需求：

功能	Bedrock Agent 使用案例	客服人員建置器
用途	叫用預先部署的 Amazon Bedrock 代理程式	建置、部署和管理自訂代理程式
組態	僅限客服人員 ID 和別名 ID	完整代理程式組態：系統提示、模型、MCP 伺服器、記憶體
部署	簡單調用層	在 AgentCore 執行期上完成代理程式生命週期
執行時間	Amazon Bedrock Agents 服務	Amazon Bedrock AgentCore 搭配 Strands SDK
工具整合	在 Bedrock Agents 主控台中設定	模型內容通訊協定 (MCP) 伺服器和內建 Strands 工具
記憶體	由 Bedrock 代理程式管理（最多 30 天）	具有可設定短期和長期保留的 AgentCore 記憶體
自訂	僅限預先部署的代理程式設定	完全控制提示、模型、工具和行為
最適合	快速部署現有的代理程式	自訂代理程式開發和生產部署

> **Note**
>
> 這兩個選項都支援即時串流、對話歷史記錄和企業級安全性。

工作流程建置器

工作流程建置器透過建立將工作委派給專業客服人員建置器客服人員的主管客服人員來啟用多重客服人員協調。每個工作流程包含：

- 主管客服人員：接收使用者請求並協調專業客服人員的進入點客服人員
- 專門客服人員：客服人員建置器使用案例，主管可以將任務委派給
- 客服人員為工具模式：主管會將每個客服人員建置器客服人員註冊為工具，並自動選取要使用的客服人員

功能	客服人員建置器	工作流程建置器
用途	建置和部署單一自訂代理程式	協調多個客服人員建置器客服人員
代理程式類型	具有 MCP 工具的單一代理程式	主管客服人員 + 多個客服人員建置器客服人員
工具整合	MCP 伺服器和 Strands 工具	登錄為工具的客服人員建置器客服人員
委派	直接工具調用	自動代理程式選擇和委派
複雜性	單一代理程式任務	多步驟、多代理程式工作流程
客服人員重複使用	N/A	重複使用現有的 代理程式建置器部署
最適合	聚焦的單一網域任務	需要多個專業化的複雜工作流程

Note

- 工作流程需要至少 1 個客服人員建置器使用案例，做為專門的客服人員
- 所有專業客服人員都必須是部署在 GAAB 中的客服人員建置器使用案例

使用案例

企業資料的問題回答

LLMs 和其他基礎模型已在大量資料組合上預先訓練，使其能夠在許多自然語言處理 (NLP) 任務中表現良好。但是，大多數的基礎模型和 LLMs都是靜態的，並且已預先訓練，限制其準確回答新、專門或專屬主題問題的能力。使用提示式學習，您可以利用 LLM 的強大 NLP 和文字產生功能，為企業資料提供更豐富的客戶體驗。

快速生成式 AI 原型設計

解決方案開箱即用，隨附各種模型提供者和使用案例。透過易於使用的部署精靈，客戶可以部署預先建置的使用案例，以快速實驗不同的生成式 AI 原型和工作負載。

多 LLM 比較和實驗

LLMs的執行方式不同，而且根據應用程式的特定需求，您可能會發現一個 LLM 比另一個 LLM 更適合您的應用程式。這可能是有關效能、準確性、成本、創造力或許多其他因素的原因。此解決方案可讓您快速部署多個使用案例，讓您能夠實驗和比較不同的組態，直到您找到符合您需求的組態為止。

概念和定義

本節說明關鍵概念並定義此解決方案特有的術語：

管理員使用者

在本指南的內容中，管理員使用者負責管理部署中包含的內容。此使用者可存取部署儀表板 UI，主要負責策劃業務使用者體驗。這是我們的主要目標客戶。

商業使用者

在本指南的內容中，商業使用者代表已部署使用案例的個人。他們是知識庫的消費者，也是負責評估和實驗 LLMs的客戶。

部署儀表板

部署儀表板是一種 Web 界面，可做為管理主控台，供管理員使用者檢視、管理和建立其使用案例。此儀表板可讓客戶利用 LLMs 快速實驗、迭代和生產各種 AI/ML 工作負載。

DevOps 使用者

在本指南的內容中，DevOps 使用者負責在 AWS 帳戶中部署解決方案，以及管理基礎設施、更新解決方案、監控效能，以及維護解決方案的整體運作狀態和生命週期。

使用案例

使用案例是獨立於與 LLMs整合的整體解決方案的應用程式，透過在新的或現有的應用程式中加入自然語言界面，來實現更豐富的客戶體驗。使用案例可透過部署儀表板或自行部署。

> **Note**
>
> 如需 AWS 術語的一般參考，請參閱 AWS 詞彙表。

架構概觀

本節提供使用此解決方案所部署元件的參考實作架構圖。

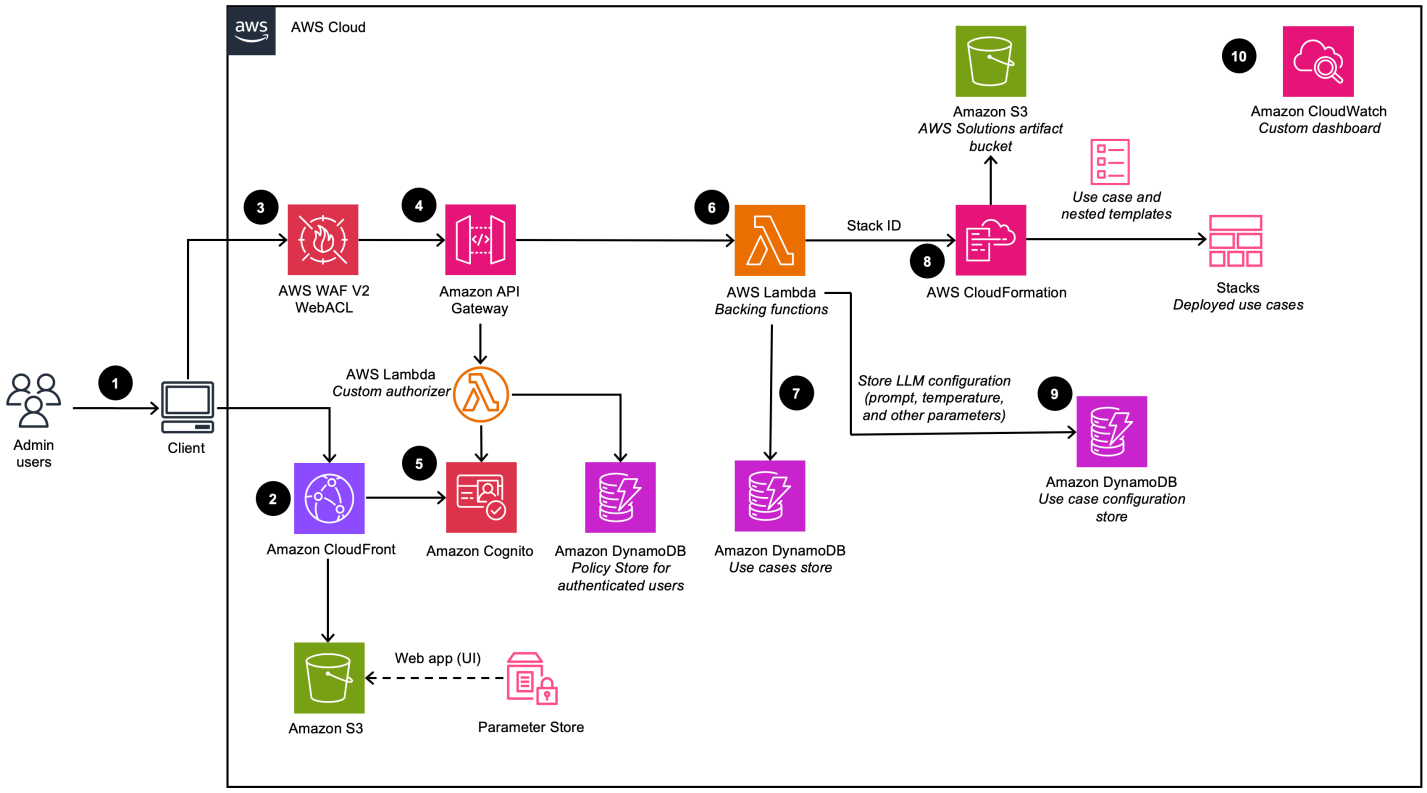
架構圖

為了支援多個使用案例和業務需求，此解決方案提供六個 AWS CloudFormation 範本：

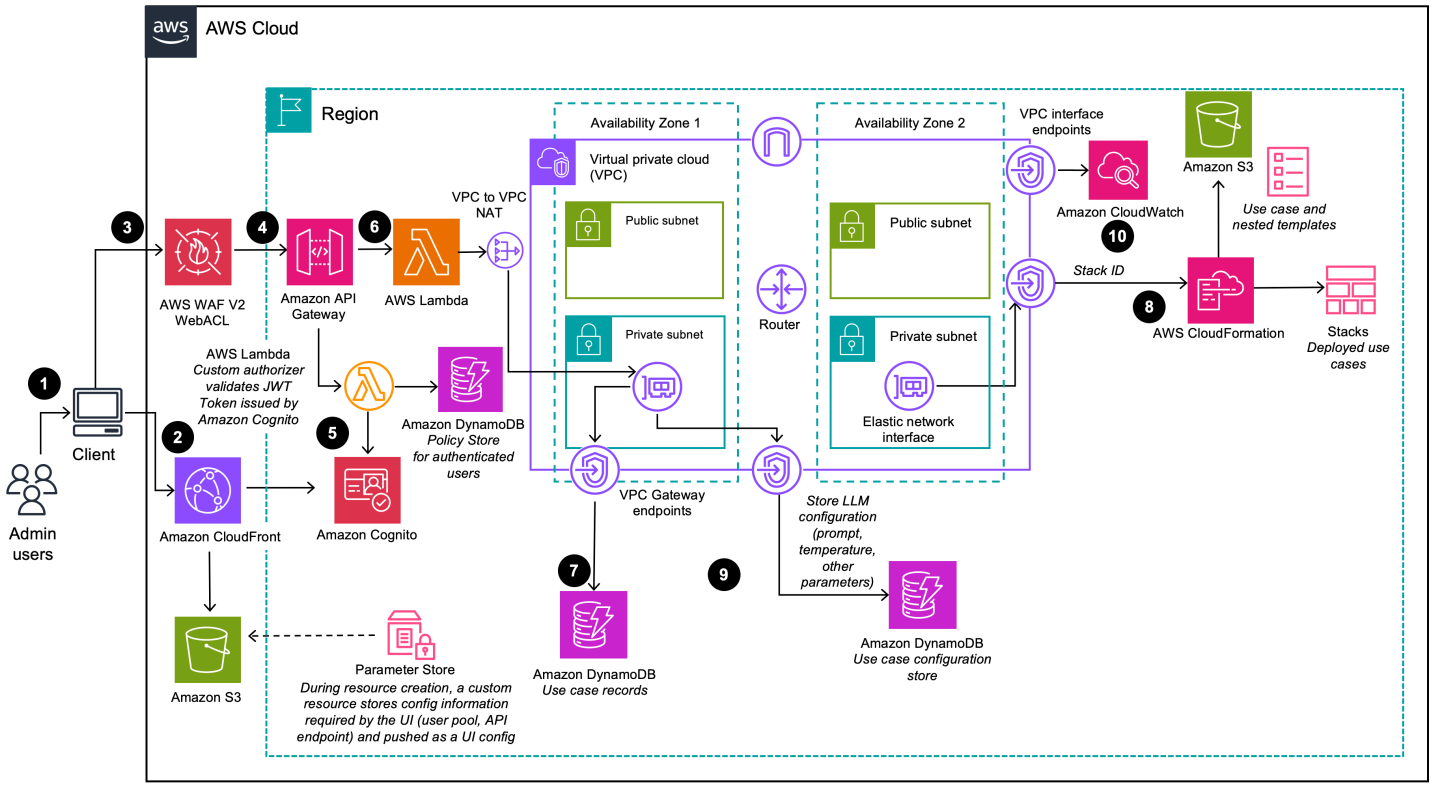
1. 部署儀表板 - 部署儀表板是一種 Web 界面，可做為管理使用者檢視、管理和建立其使用案例的管理主控台。此儀表板可讓客戶利用 LLMs 快速實驗、迭代和生產各種 AI/ML 工作負載。
2. 文字使用案例 - 文字使用案例可讓使用者使用生成式 AI 體驗自然語言界面。此使用案例可以整合到新的或現有的應用程式，並且可以透過部署儀表板或透過提供的 URL 獨立部署。
3. Bedrock 代理程式使用案例 - Bedrock 代理程式使用案例可讓現有的 Bedrock 代理程式完成任務或自動化重複的工作流程。
4. MCP 伺服器 - MCP 伺服器使用案例可啟用模型內容通訊協定伺服器的部署和管理，以提供 AI 應用程式的標準化工具和資源存取。支援封裝現有 Lambda 函數、APIs 和外部 MCP 伺服器的閘道方法，以及部署自訂容器化 MCP 伺服器的執行期方法。
5. 代理程式建置器 - 代理程式建置器可透過完整的組態控制、MCP 伺服器整合和記憶體管理功能，在 Amazon Bedrock AgentCore 上建立和部署生產就緒 AI 代理程式。
6. 工作流程建置器 - 工作流程建置器可建立主管客服人員，使用客服人員做為工具委派模式來協調多個客服人員建置器客服人員，以處理複雜的多客服人員工作流程。

部署儀表板

描述部署儀表板架構 （在停用 VPC 選項的情況下部署時）



描述部署儀表板架構 （ 在啟用 VPC 選項的情況下部署時 ）



> **Note**
>
> AWS CloudFormation 資源是從 AWS 雲端開發套件 (AWS CDK) 建構模組建立。

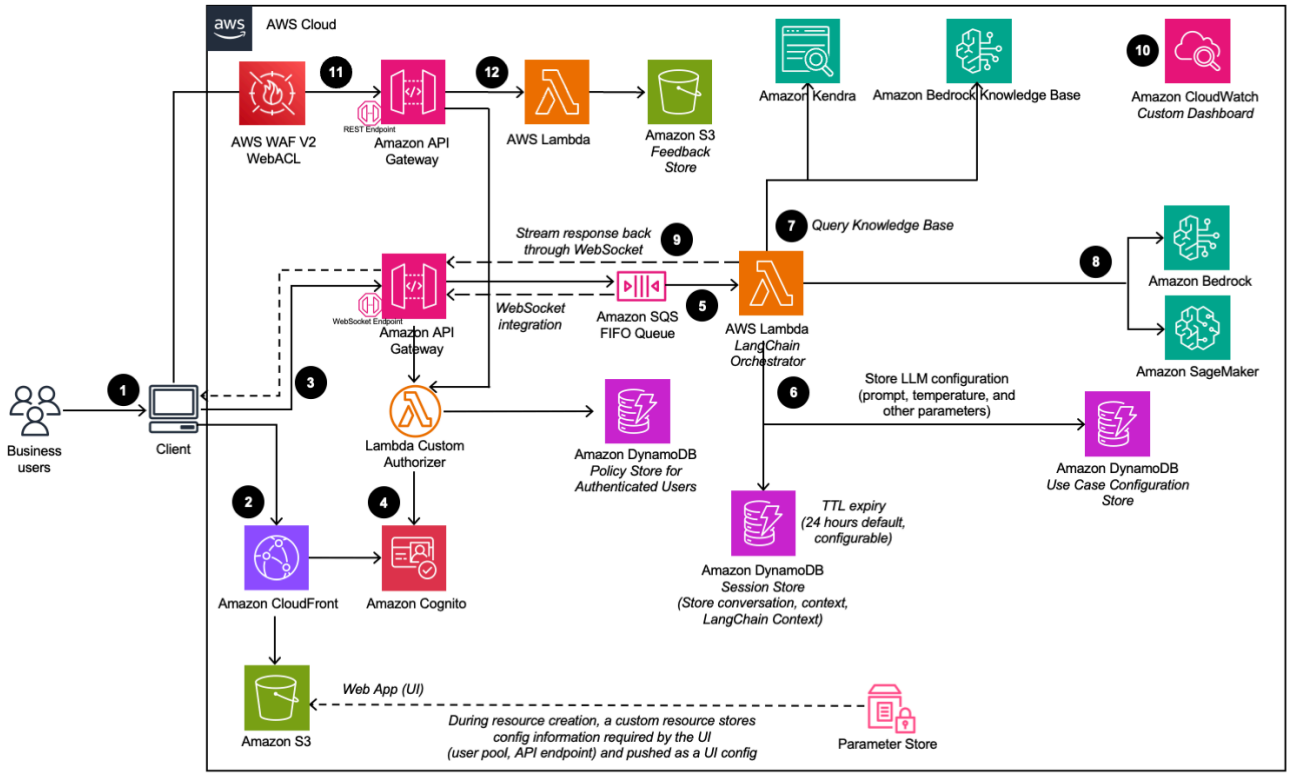
使用 AWS CloudFormation 範本部署之解決方案元件的高階程序流程如下：

1. 管理員使用者登入部署儀表板使用者介面 (UI)。
2. Amazon CloudFront 提供 Web UI，其託管在 Amazon Simple Storage Service (Amazon S3) 儲存貯體中。
3. AWS WAF 可保護 APIs免受攻擊。此解決方案會設定一組稱為 Web 存取控制清單 (Web ACL) 的規則，以根據可設定的、使用者定義的 Web 安全規則和條件來允許、封鎖或計數 Web 請求。
4. Web UI 會利用一組使用 Amazon APIs的 REST API。 Amazon API Gateway
5. Amazon Cognito 會驗證使用者，並傳回 CloudFront Web UI 和 API Gateway。
6. AWS Lambda 為 REST 端點提供商業邏輯。此備份 Lambda 函數會管理和建立必要的資源，以使用 AWS CloudFormation 執行使用案例部署。
7. Amazon DynamoDB 會存放部署清單。
8. 當管理員使用者建立新的使用案例時，後端 Lambda 函數會為請求的使用案例啟動 CloudFormation 堆疊建立事件。
9. 部署精靈中管理員使用者提供的所有 LLM 組態選項都會儲存在 DynamoDB 中。部署使用此 DynamoDB 資料表在執行時間設定 LLM。
10. 使用 Amazon CloudWatch，此解決方案會從各種服務收集操作指標，以產生自訂儀表板，讓您監控解決方案的效能和運作狀態。

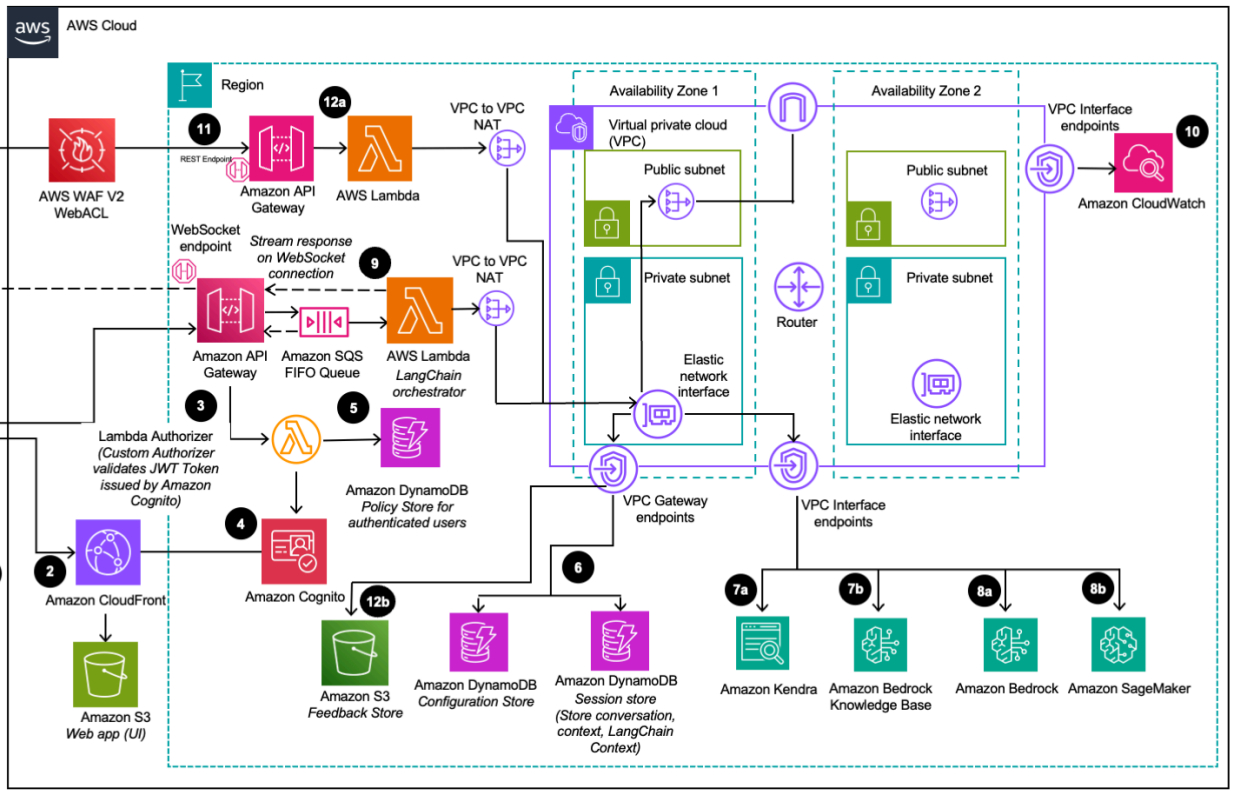
> **Note**
>
> - 如果您選擇在 Amazon VPC 中部署此解決方案，資料將在您的私有網路中路由。
> - 雖然可在大多數 AWS 區域中啟動部署儀表板，但已部署的使用案例會根據服務可用性而有特定限制。如需詳細資訊，請參閱支援的 AWS 區域。

文字使用案例

描述文字使用案例架構（在停用 VPC 選項的情況下部署時）



描述文字使用案例架構（在啟用 VPC 選項的情況下部署時）



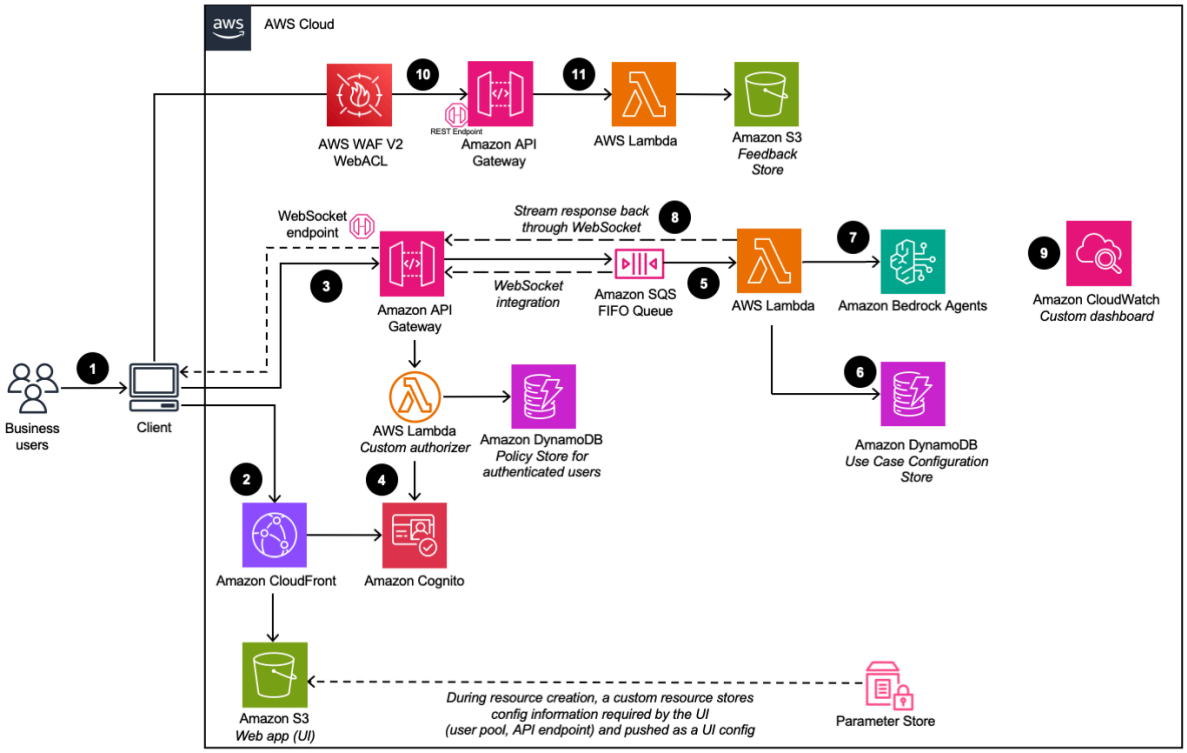
使用 AWS CloudFormation 範本部署之解決方案元件的高階程序流程如下：

1. 管理員使用者使用 部署儀表板部署使用案例。商業使用者登入使用案例 UI。
2. CloudFront 提供託管在 S3 儲存貯體中的 Web UI。
3. Web UI 利用使用 API Gateway 建置的 WebSocket 整合。API Gateway 由自訂 Lambda 授權方函數提供支援，該函數會根據驗證使用者所屬的 Amazon Cognito 群組，傳回適當的 AWS Identity and Access Management (IAM) 政策。此政策存放在 DynamoDB 中。
4. Amazon Cognito 會驗證使用者，並傳回 CloudFront Web UI 和 API Gateway。
5. 來自商業使用者的傳入請求會從 API Gateway 傳遞至 Amazon SQS 佇列，然後傳遞至 LangChain Orchestrator。LangChain Orchestrator 是 Lambda 函數和 layer 的集合，可提供商業邏輯以滿足來自商業使用者的請求。佇列會啟用 API Gateway 與 Lambda 整合的非同步操作。佇列會將連線資訊傳遞至 Lambda 函數，然後直接將結果發佈回 API Gateway Websocket 連線，以支援長時間執行的推論呼叫。
6. LangChain Orchestrator 使用 Amazon DynamoDB 取得設定的 LLM 選項和必要的工作階段資訊（例如聊天歷史記錄）。
7. 如果部署已啟用知識庫，則 LangChain Orchestrator 會利用 Amazon Kendra 或 Amazon Bedrock 的知識庫來執行搜尋查詢以擷取文件摘錄。
8. 使用知識庫的聊天歷史記錄、查詢和內容，LangChain Orchestrator 會建立最終提示，並將請求傳送至 Amazon Bedrock 或 Amazon SageMaker AI 上託管的 LLM。
9. 當回應從 LLM 傳回時，LangChain Orchestrator 會透過用戶端應用程式要使用的 API Gateway WebSocket 串流回回應。
10. 使用 Amazon CloudWatch，此解決方案會從各種服務收集操作指標，以產生自訂儀表板，讓您監控部署的效能和運作狀態。
11. 如果啟用意見回饋收集，則可利用利用 Amazon API Gateway 的 REST API 端點來收集使用者意見回饋。
12. 意見回饋後端 lambda、使用其他使用案例特定的中繼資料（例如使用的模型）來增強提交的意見回饋，並將資料存放在 Amazon S3 中，以供 DevOps 使用者日後分析和報告。

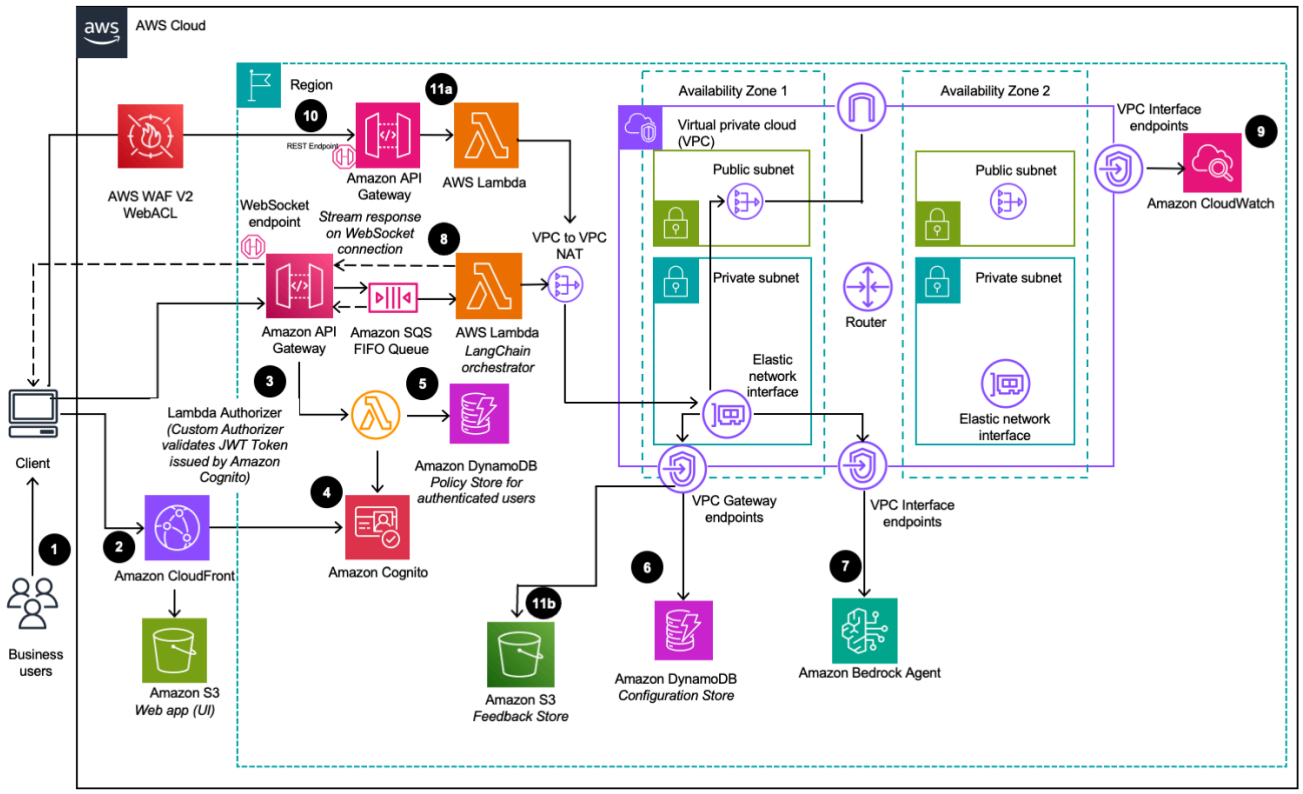
> **Note**
>
> 如果您選擇在 Amazon VPC 中部署此解決方案，資料將路由到您的私有網路。

Bedrock Agent 使用案例

描述 Bedrock Agent 使用案例架構 （ 在停用 VPC 選項的情況下部署時 ）



描述 Bedrock Agent 使用案例架構 （ 在啟用 VPC 選項的情況下部署時 ）



使用 AWS CloudFormation 範本部署之解決方案元件的高階程序流程如下：

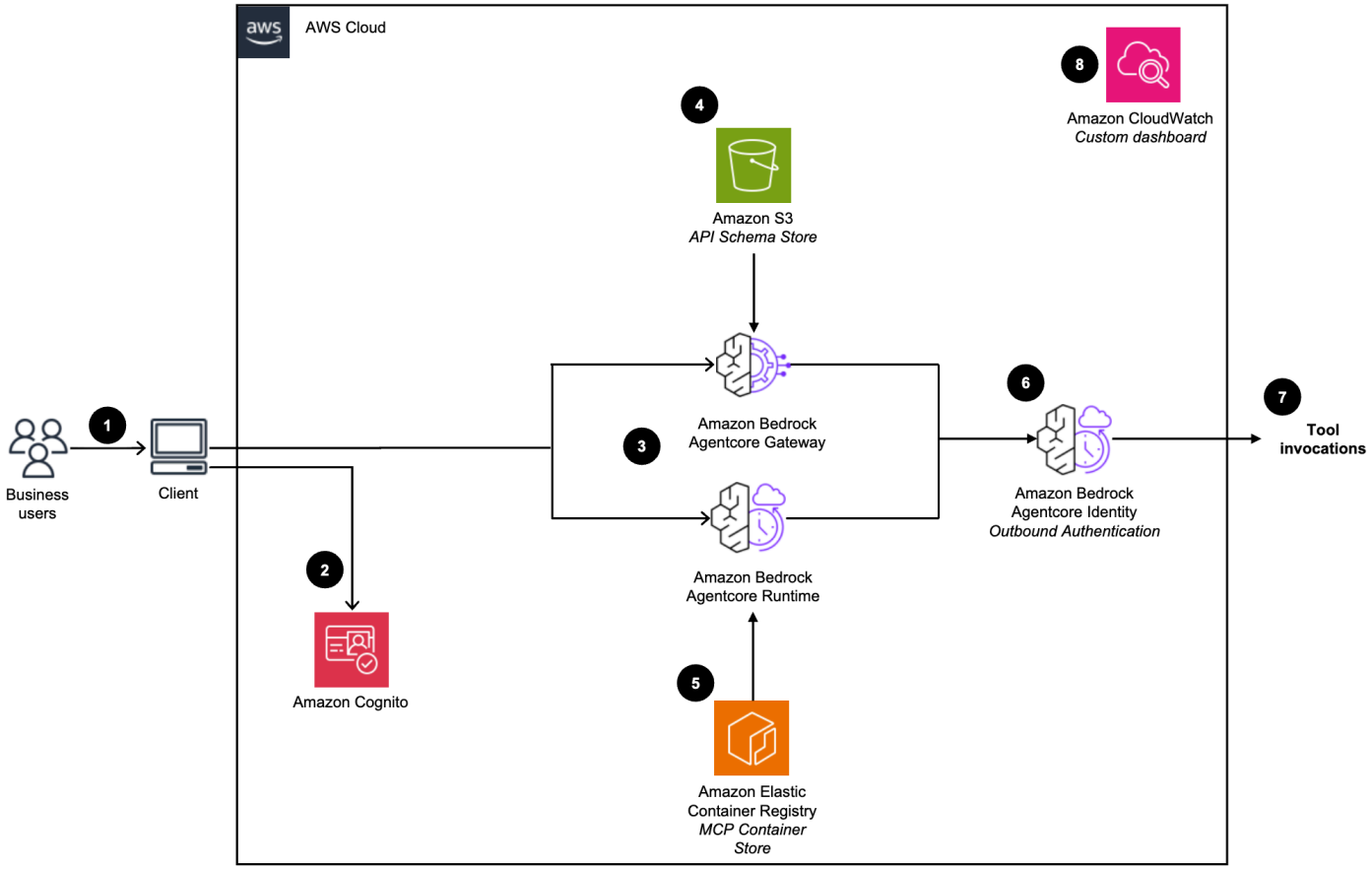
1. 管理員使用者使用 部署儀表板部署使用案例。商業使用者登入使用案例 UI。
2. CloudFront 提供託管在 S3 儲存貯體中的 Web UI。
3. Web UI 利用使用 API Gateway 建置的 WebSocket 整合。API Gateway 由自訂 Lambda 授權方函數提供支援，該函數會根據驗證使用者所屬的 Amazon Cognito 群組，傳回適當的 AWS Identity and Access Management(IAM) 政策。此政策存放在 DynamoDB 中。
4. Amazon Cognito 會驗證使用者，並傳回 CloudFront Web UI 和 API Gateway。
5. 來自商業使用者的傳入請求會從 API Gateway 傳遞至 Amazon SQS 佇列，然後傳遞至 AWS Lambda 函數。佇列會啟用 API Gateway 與 Lambda 整合的非同步操作。佇列會將連線資訊傳遞至 Lambda 函數，然後直接將結果發佈回 API Gateway Websocket 連線，以支援長時間執行的推論呼叫。
6. AWS Lambda 函數使用 Amazon DynamoDB 視需要取得使用案例組態
7. 使用使用者輸入和任何相關的使用案例組態，AWS Lambda 函數會建置請求承載並傳送至設定的 Amazon Bedrock Agent，以滿足使用者意圖。
8. 當回應從 Amazon Bedrock 代理程式傳回時，Lambda 函數會透過用戶端應用程式要使用的 API Gateway WebSocket 串流回回應。

9. 使用 Amazon CloudWatch，此解決方案會從各種服務收集操作指標，以產生自訂儀表板，讓您監控部署的效能和運作狀態。
10. 如果啟用意見回饋收集，則可利用利用 Amazon API Gateway 的 REST API 端點來收集使用者意見回饋。
11. 回饋支援 Lambda，使用額外的使用案例特定中繼資料來增強提交的回饋，並將資料存放在 Amazon S3 中，以供 DevOps 使用者日後分析和報告。

> **Note**
>
> 如果您選擇在 Amazon VPC 中部署此解決方案，資料將在您的私有網路中路由。

MCP 伺服器使用案例

描述 MCP Server 使用案例架構



MCP Server 使用案例可在 Amazon Bedrock AgentCore 上部署和管理模型內容通訊協定伺服器。MCP 伺服器為 AI 應用程式提供標準化界面，以存取工具、資源和企業資料來源。

解決方案支援兩種部署方法：

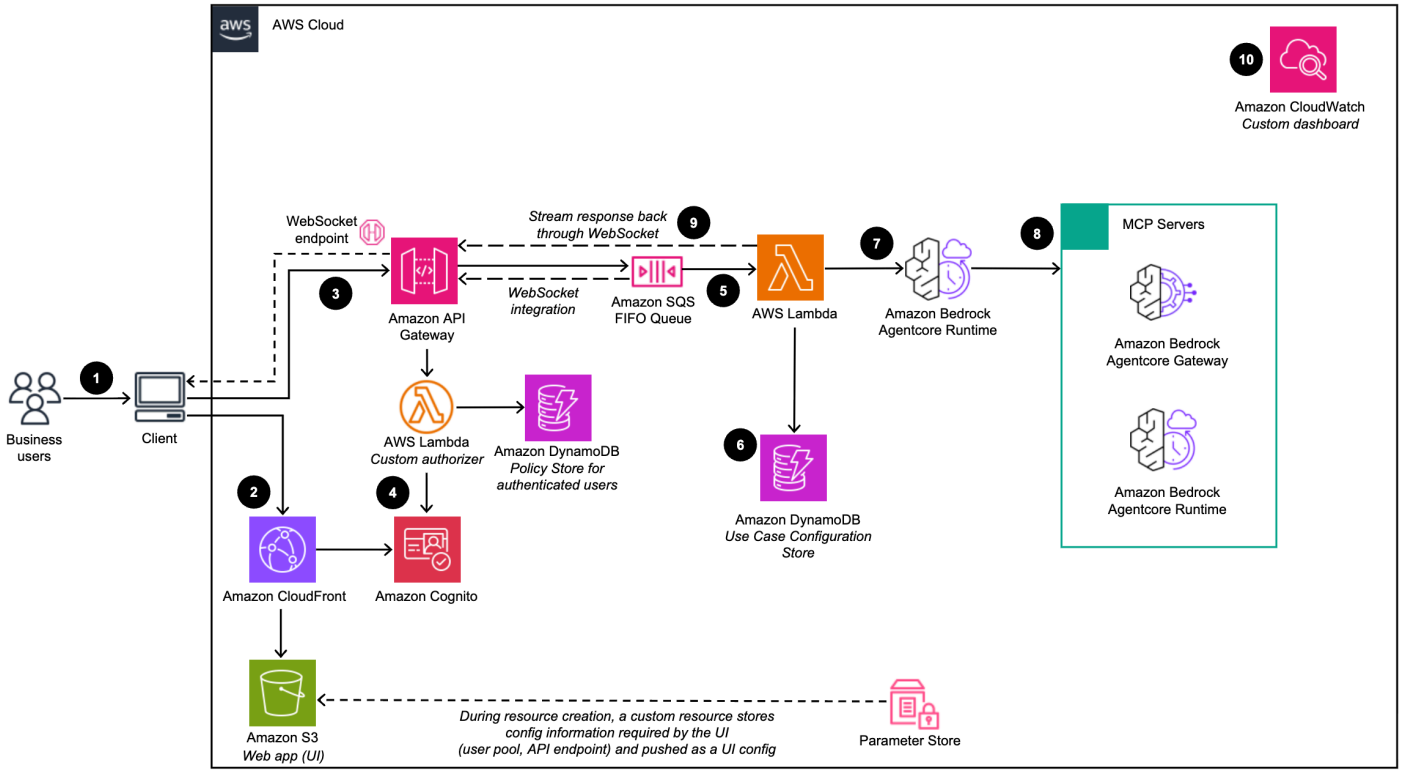
- 閘道方法：將現有的 Lambda 函數、REST APIs 或外部 MCP 伺服器包裝為 MCP 工具，自動處理通訊協定轉譯
- 執行期方法：從 Amazon ECR 映像部署自訂容器化 MCP 伺服器

MCP Server 部署的高階程序流程如下：

1. 管理員使用者使用部署儀表板部署 MCP 伺服器使用案例，選取閘道或執行期部署方法。
2. 此動作使用 Amazon Cognito 驗證。
3. 對於閘道部署，解決方案會建立 Amazon Bedrock AgentCore Gateway，將現有的 Lambda 函數、APIs 或外部 MCP 伺服器轉換為 MCP 相容工具。對於執行期部署，解決方案會使用提供的 ECR 映像，在 Amazon Bedrock AgentCore 執行期部署容器化 MCP 伺服器。
4. 閘道部署會從 Amazon S3 中上傳的位置擷取必要的 API/Lambda/Smithy 結構描述，或直接連線至 MCP Server URL 端點。
5. 執行期部署會從 Amazon Elastic Container Registry (ECR) 擷取使用者提供的容器化 MCP 伺服器
6. MCP 伺服器使用 Amazon Bedrock AgentCore Identity OAuth 用戶端進行檢測
7. MCP Server 會在 /mcp 端點提供相關的工具，以供客服人員探索。
8. Amazon CloudWatch 會從 MCP 伺服器部署收集操作指標和日誌，以進行監控和故障診斷。

客服人員建置器使用案例

Depicts Agent Builder 架構



使用 AWS CloudFormation 範本部署之 Agent Builder 元件的高階程序流程如下：

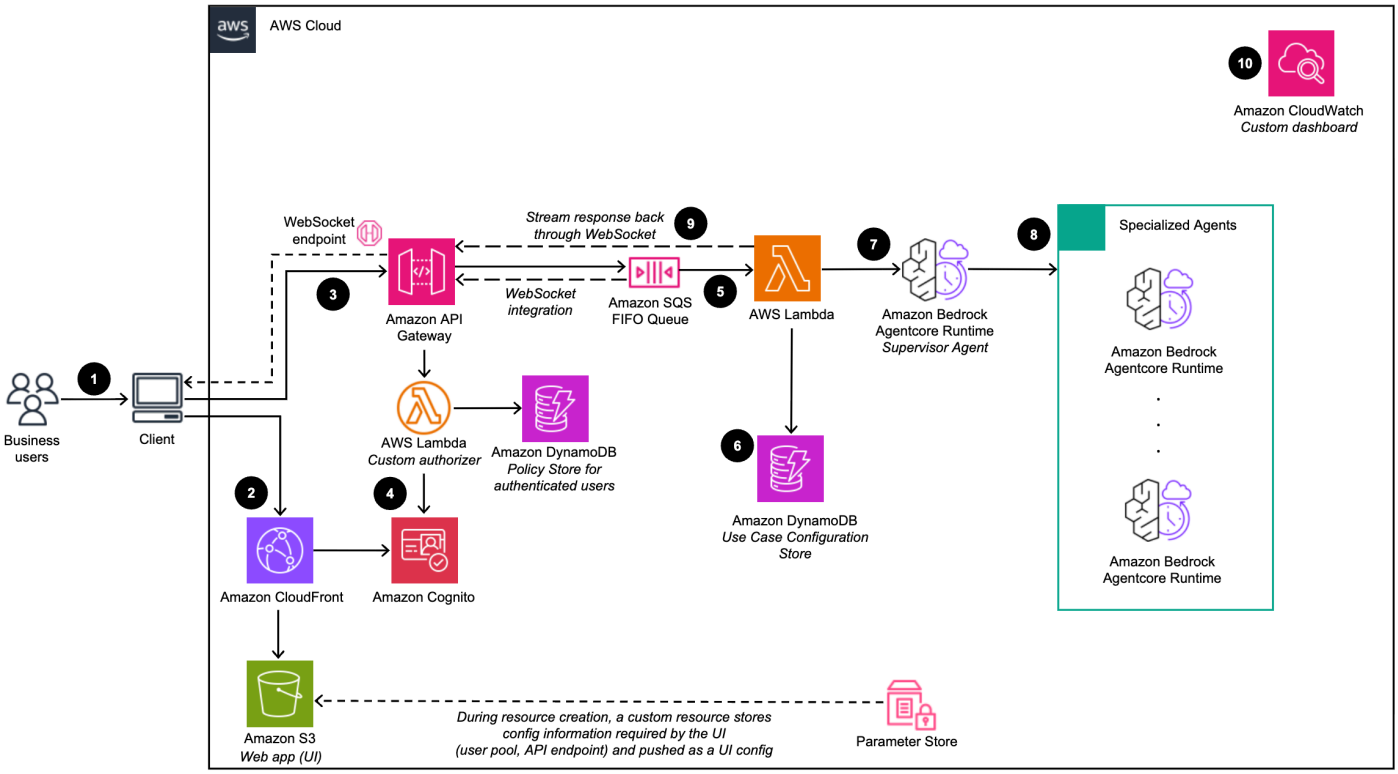
1. 管理員使用者使用 部署儀表板部署使用案例。商業使用者登入使用案例 UI。
2. CloudFront 提供託管在 S3 儲存貯體中的 Web UI。
3. Web UI 利用使用 API Gateway 建置的 WebSocket 整合。API Gateway 由自訂 Lambda 授權方函數提供支援，該函數會根據驗證使用者所屬的 Amazon Cognito 群組，傳回適當的 AWS Identity and Access Management(IAM) 政策。此政策存放在 DynamoDB 中。
4. Amazon Cognito 會驗證使用者，並傳回 CloudFront Web UI 和 API Gateway。
5. 來自商業使用者的傳入請求會從 API Gateway 傳遞至 Amazon SQS 佇列，然後傳遞至 AWS Lambda 函數。佇列會啟用 API Gateway 與 Lambda 整合的非同步操作。佇列會將連線資訊傳遞至 Lambda 函數，然後直接將結果發佈回 API Gateway Websocket 連線，以支援長時間執行的推論呼叫。
6. AWS Lambda 函數會從 DynamoDB 擷取代理程式組態。
7. 使用使用者輸入和任何相關的使用案例組態，AWS Lambda 函數會建置請求承載，並將其傳送至在 Amazon Bedrock AgentCore 執行期上執行的代理程式。
8. 代理程式會連線至相關聯的 MCP 伺服器，並將工具註冊到 strands 代理程式執行個體。代理程式接著會根據工具描述和任務需求自動選取和執行動作。

9. 當回應從 Amazon Bedrock AgentCore 執行時間傳回時，Lambda 函數會透過用戶端應用程式要使用的 API Gateway WebSocket 串流回回應。

> **Note**
>
> - 代理程式處理僅限於 Lambda 執行逾時 (15 分鐘）。

工作流程建置器使用案例

描述工作流程建置器架構



使用 AWS CloudFormation 範本部署之工作流程建置器元件的高階程序流程如下：

1. 管理員使用者使用部署儀表板部署工作流程，選取 Agent Builder 代理程式以包含為專門的代理程式。
2. CloudFront 提供託管在 S3 儲存貯體中的 Web UI。

3. Web UI 利用使用 API Gateway 建置的 WebSocket 整合。API Gateway 由自訂 Lambda 授權方函數提供支援，該函數會根據驗證使用者所屬的 Amazon Cognito 群組，傳回適當的 AWS Identity and Access Management (IAM) 政策。此政策存放在 DynamoDB 中。
4. Amazon Cognito 會驗證使用者，並傳回 CloudFront Web UI 和 API Gateway。
5. 來自商業使用者的傳入請求會從 API Gateway 傳遞至 Amazon SQS 佇列，然後傳遞至 AWS Lambda 函數。佇列會啟用 API Gateway 與 Lambda 整合的非同步操作。
6. AWS Lambda 函數會從 DynamoDB 擷取工作流程組態，包括專用 Agent Builder 代理程式的清單。
7. 使用使用者輸入和工作流程組態，Lambda 會將請求傳送至託管主管代理程式的 Amazon Bedrock AgentCore 執行期。
8. 主管代理程式會在 AgentCore 執行期環境中建立所有專業 Agent Builder 代理程式的本機執行個體。這些專門代理程式會使用 代理程式做為工具模式註冊為工具。然後，主管會根據客服人員描述和任務需求，自動選取並將工作委派給專門的客服人員。
9. 主管客服人員會彙總專業客服人員的結果，並制定最終回應，將其傳回至 Lambda，以便透過 API Gateway Websocket 串流回用戶端應用程式。

> **Note**
>
> - 工作流程處理僅限於 Lambda 執行逾時 (15 分鐘）。

AWS Well-Architected 設計考量事項

此解決方案的設計採用 AWS Well-Architected Framework 的最佳實務，可協助客戶在雲端中設計和操作可靠、安全、高效且符合成本效益的工作負載。

本節說明如何在建置此解決方案時套用 Well-Architected Framework 的設計原則和最佳實務。

卓越營運

本節說明如何使用卓越營運支柱的原則和最佳實務來建構此解決方案。

- 我們使用 Amazon CloudFormation 將解決方案建置為infrastructure-as-code。
- Lambda 函數會將自訂指標推送至 CloudWatch 和自訂 CloudWatch 儀表板，以監控解決方案的運作狀態。
- 解決方案元件經過高度模組化，可讓您靈活選擇要部署的元件。

安全

本節說明如何使用安全支柱的原則和最佳實務來建構此解決方案。

- 部署儀表板和所有使用案例都會使用 Amazon Cognito 進行身分驗證和授權。
- 所有服務間通訊都使用 AWS IAM 角色。
- 所有解決方案角色都遵循最低權限存取；也就是說，只會授予所需的最低許可。
- 所有資料儲存，包括 S3 儲存貯體、DynamoDB 和 Amazon Kendra 都有靜態加密。

可靠性

本節說明如何使用可靠性支柱的原則和最佳實務來建構此解決方案。

- 以無伺服器範例為基礎的架構。
- 我們建置了隨需、水平可擴展性和從基礎基礎設施故障自動復原的架構。
- 架構包括緩衝和限流請求，以免造成基礎端點過載。

效能效率

本節說明如何使用效能效率支柱的原則和最佳實務來建構此解決方案。

- 解決方案使用 DynamoDB，這是一種全受管無伺服器 NoSQL 資料庫，具有隨需擴展功能。
- 解決方案使用 Amazon S3 進行物件儲存和託管網站 （透過 CloudFront)，以 11 個 9s 耐用性提供低成本、可擴展性。

成本最佳化

本節說明如何使用成本最佳化支柱的原則和最佳實務來建構此解決方案。

- 在可能的情況下，我們建置解決方案以使用無伺服器架構；因此您只需支付使用量的費用。

永續性

本節說明如何使用永續性支柱的原則和最佳實務來建構此解決方案。

- 解決方案的模組化元件化架構可讓您靈活地自訂要針對個別使用案例佈建的資源。

- 架構使用無伺服器運算和儲存，可最佳化資源使用率。
- 作為雲端解決方案，此解決方案受益於共用資源、聯網、電源冷卻和實體設施。

架構詳細資訊

本節說明構成此解決方案的元件和 AWS 服務，以及這些元件如何一起運作的架構詳細資訊。

此解決方案中的 AWS 服務

AWS 服務	Description
Amazon API Gateway	核心。此服務提供部署儀表板的 REST APIs，以及使用案例的 WebSocket API。
AWS CloudFormation	核心。此解決方案以 CloudFormation 範本的形式分佈，而 CloudFormation 會部署解決方案的 AWS 資源。
Amazon CloudFront	核心。CloudFront 提供 Amazon S3 中託管的 Web 內容。
Amazon Cognito	核心。此服務會處理 API 的使用者管理和身分驗證。
Amazon DynamoDB	核心。DynamoDB 會存放部署儀表板的部署資訊和組態詳細資訊。它將聊天歷史記錄和對話 IDs存放在文字使用案例中，以啟用對話歷史記錄和查詢歧義。
AWS Lambda	核心。解決方案使用 Lambda 函數來： * 返回 REST 和 WebSocket API 端點 * 處理每個使用案例協調器的核心邏輯 * 在 CloudFormation 部署期間實作自訂資源
Amazon S3	核心。Amazon S3 託管靜態 Web 內容。
Amazon CloudWatch	支援。此解決方案會將日誌從解決方案資源發佈至 CloudWatch Logs，並將指標發佈至 CloudWatch 指標。解決方案也會建立 CloudWatch 儀表板來檢視此資料。

AWS 服務	Description
AWS Systems Manager	支援。Systems Manager 提供資源操作和成本資料的應用程式層級資源監控和視覺化。也用於在參數存放區中存放組態資料。
AWS WAF	支援。AWS WAF 部署在 API Gateway 部署之前，以保護它。
Amazon Bedrock	「選用」。解決方案利用 Amazon Bedrock 存取基礎或自訂模型、Amazon Bedrock Agents、Amazon Bedrock 知識庫。Amazon Bedrock 是建議整合，可防止您的資料離開 AWS 網路。
Amazon Bedrock AgentCore	選用 解決方案可控制 Amazon Bedrock AgentCore 執行和支援 MCP Server 連線，以及 Agent Builder 和 Workflow 使用案例。
Amazon Elastic Container Registry (Amazon ECR)	「選用」。對於代理程式建置器部署，ECR 會存放和分發代理程式容器映像。解決方案使用 ECR Pull-Through Cache 自動從 GAAB 團隊公有 ECR 儲存庫擷取預先建置的代理程式映像。
AWS Distro for OpenTelemetry (ADOT)	「選用」。對於客服人員建置器部署，ADOT 提供客服人員可觀測性的自動檢測，為客服人員操作啟用分散式追蹤和結構化記錄。
Amazon Kendra	「選用」。在文字使用案例中，管理員使用者可以選擇性地決定連接 Amazon Kendra 索引，以用作與 LLM 對話的知識庫。這可用於將新資訊插入 LLM，使其能夠在其回應中使用該資訊。

AWS 服務	Description
Amazon SageMaker AI	「選用」。解決方案可與 Amazon SageMaker AI 推論端點整合，以存取 AWS 帳戶和區域內託管FMs，並且是防止資料離開 AWS 網路的偏好整合。 **Note** 您必須在推論端點可用的相同區域中部署解決方案。
Amazon Virtual Private Cloud	「選用」。解決方案提供使用啟用 VPC 的組態部署元件的選項。使用已啟用 VPC 的組態部署解決方案時，您可以選擇讓解決方案為您建立 VPC，或使用存在於部署解決方案之相同帳戶和區域中的現有 VPC （使用自有 VPC)。如果解決方案建立 VPC，它會建立必要的網路元件，包括子網路、安全群組及其規則、路由表、網路 ACLs、NAT Gateway、網際網路閘道、VPC 端點及其政策。

部署儀表板

API Gateway 自訂授權方

在表面下方，API Gateway 的 Lambda 自訂授權方用於所有 API 呼叫 (RESTful 和 WebSocket 型），以驗證指定的使用者是否具有根據其所屬群組執行動作的許可。此自訂授權方由包含每個群組政策的 DynamoDB 資料表提供支援。叫用 API 時，API Gateway 會叫用自訂授權方 Lambda 函數，其會解碼提供的 Amazon Cognito 存取權杖，以判斷使用者所屬的使用者群組。然後，政策資料表會依群組名稱查詢，以傳回該群組的相關政策。

在每次新的使用案例部署上，管理員政策都會更新，以存放新的陳述式，允許在該使用案例的 API 上執行 execute-api：Invoke 動作。刪除使用案例時，對應的陳述式會從政策中移除。

對於為個別使用案例建立的群組，政策中僅存在單一陳述式，允許僅針對該使用案例的 API 執行 api：Invoke 動作。

由於此結構，屬於使用案例群組的任何使用者都可以存取該使用案例的 API。單一使用者也可以手動新增至多個群組，以允許該使用者使用多個使用案例。

> **Warning**
>
> 如果您想要將新使用案例的存取權授予現有使用者群組，您也可以在政策資料表中手動編輯指定群組的政策。刪除使用案例時（即使您已手動編輯），也會刪除使用案例群組，因此刪除使用案例時請小心。

如果使用案例堆疊是獨立部署（不使用部署儀表板），則會為該部署建立 Amazon Cognito 使用者集區，其中包含可存取該使用案例 API 的單一使用者。此使用者集區僅屬於此使用案例，不會與其他獨立部署共用。

文字使用案例

串流支援

在聊天應用程式中，延遲是啟用回應式使用者體驗的重要指標。LLM 推論可能需要幾秒鐘到幾分鐘的時間，在如何為客戶提供最佳內容服務方面帶來挑戰。因此，數個 LLM 提供者允許將回應串流回發起人。在傳回回應之前，您可以先傳回每個字符，而不是等待整個推論完成。

為了支援使用此功能，文字使用案例旨在使用 WebSocket API 來支援聊天體驗。此 WebSocket 透過 API Gateway 部署。使用 WebSocket API 可在聊天工作階段開始時建立連線，並透過該通訊端串流回應。這可讓前端應用程式提供更好的使用者體驗。

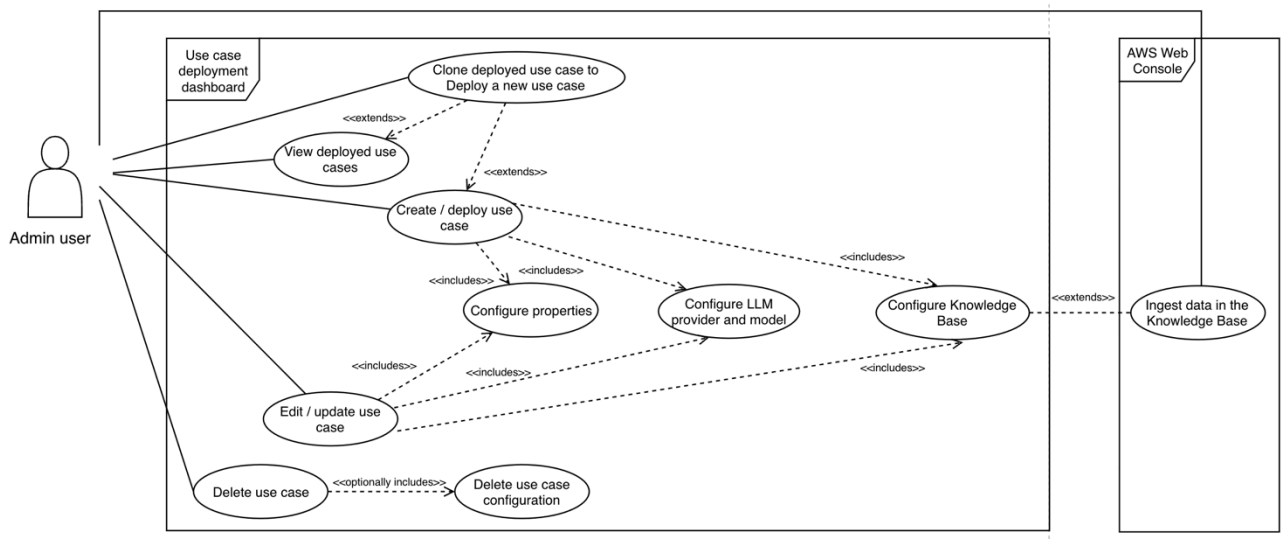
> **Note**
>
> 即使模型提供串流支援，這不一定表示解決方案將能夠透過 WebSocket API 將回應串流回去。解決方案需要啟用自訂邏輯，以支援每個模型提供者的串流。如果串流可用，管理員使用者可以在部署時間啟用/停用此功能。

AWS 解決方案上的生成式 AI 應用程式建置器如何運作

管理員使用者主要與部署儀表板互動，以檢視、建立和管理新的和現有的使用案例部署。透過此儀表板，管理員使用者可以存取下列動作：

- 檢視部署清單
- 建立新的部署
- 編輯現有的部署
- 複製部署的組態以建立新的部署
- 刪除部署 （透過 CloudFormation 刪除來取消佈建資源）
- 永久刪除部署的組態詳細資訊

描述 部署儀表板管理員使用者的使用案例圖表

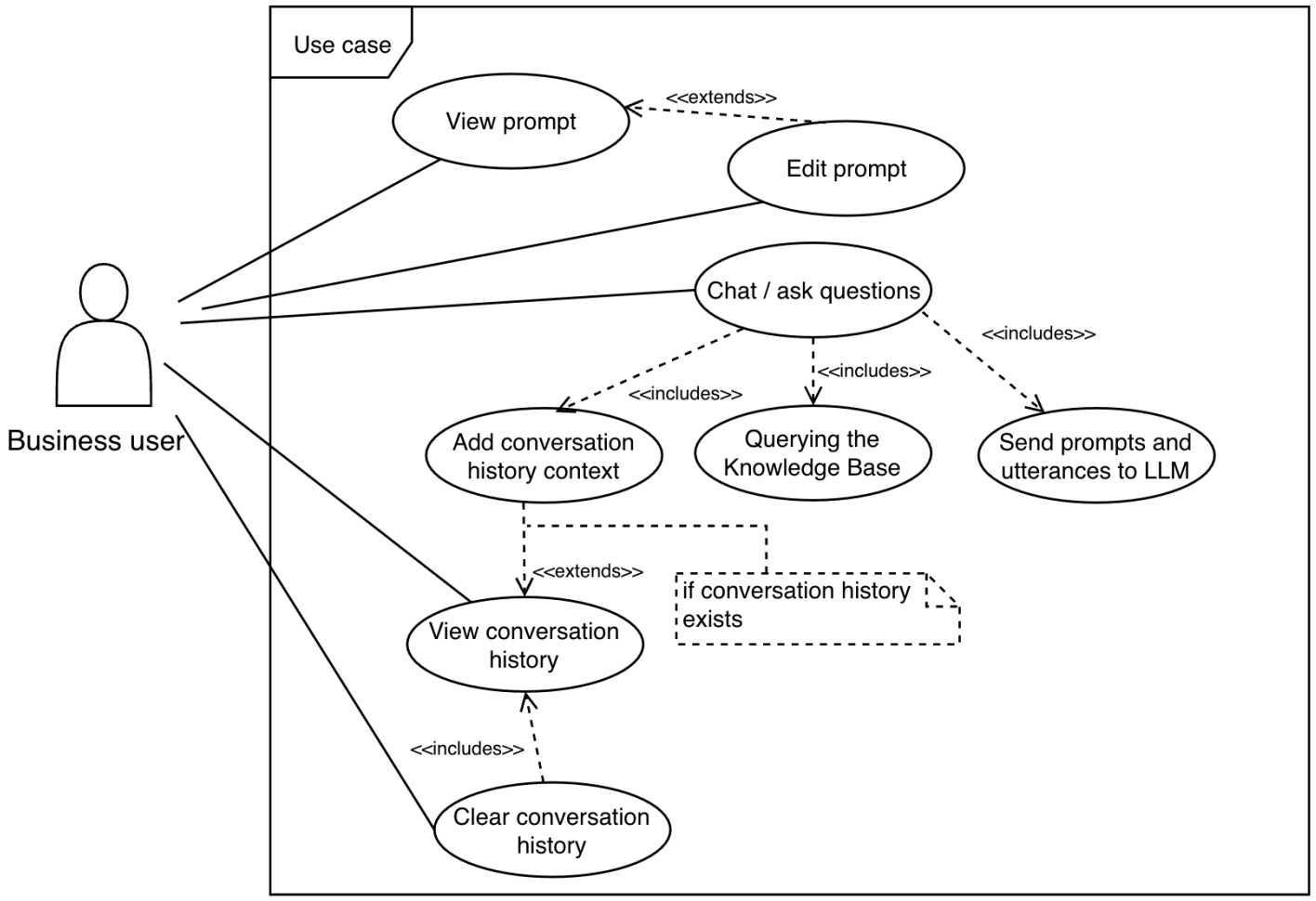


> **Note**
>
> 管理員使用者可能無法直接存取 AWS 主控台。在這種情況下，管理員使用者必須與 DevOps 使用者合作，以支援將資料擷取到 Kendra 知識庫等動作。

對於文字使用案例，商業使用者可以存取使用者介面，以便與 LLM 聊天。此組態的詳細資訊是由管理員使用者設定的部署設定所控制。在文字使用案例中，商業使用者可存取下列動作：

- 透過聊天介面傳送訊息
- 檢視對話歷史記錄
- 清除對話歷史記錄
- 檢視提示
- 編輯提示

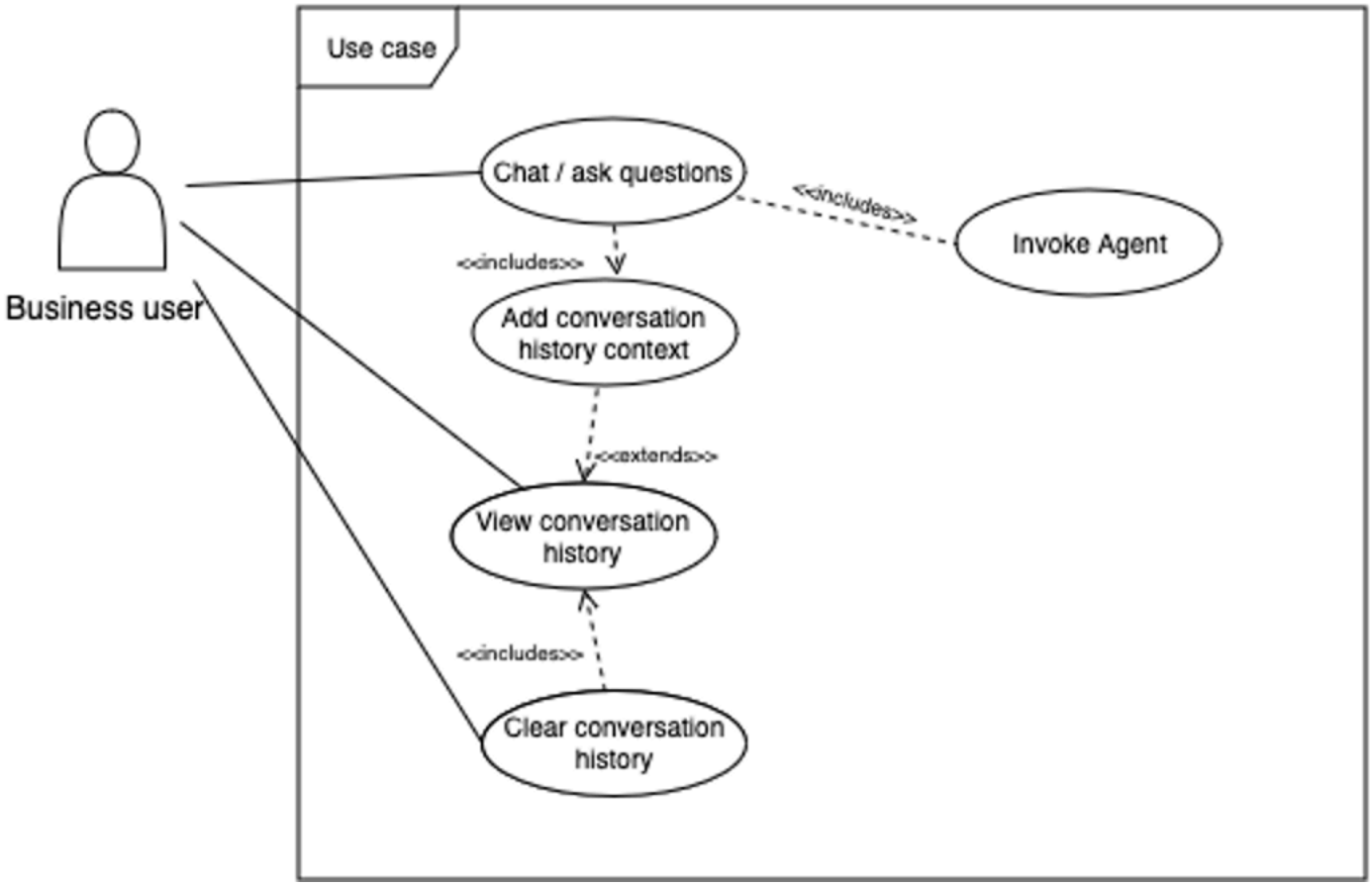
描述 文字使用案例之商業使用者的使用案例圖表



透過 Bedrock 代理程式使用案例，商業使用者可以存取 UI 與設定的 Amazon Bedrock 代理程式聊天。管理員使用者可以在部署設定中設定這些詳細資訊。在 Bedrock Agent 使用案例中，商業使用者可存取下列動作：

- 透過聊天介面傳送訊息
- 檢視對話歷史記錄
- 清除對話歷史記錄

描述 Bedrock 代理程式使用案例的業務使用者使用案例圖表



客服人員建置器

Agent Builder 提供在 Amazon Bedrock AgentCore 上建立、部署和管理生產就緒 AI 代理程式的平台。本節說明技術元件和實作詳細資訊。

AgentCore 整合

Agent Builder 使用組態型部署方法搭配預先建置的代理程式映像，以啟用快速、安全且可擴展的代理程式部署。

預先建置的代理程式映像

客服人員容器映像由 GAAB 團隊在 CI/CD 管道期間建置，並發佈至公有 ECR 儲存庫。每個映像版本都繫結至 GAAB 解決方案版本 （ 例如 v4.0.0 → gaab-strands-agent：v4.0.0)。影像是以 Strands SDK 為基礎，並包含：

- 代理程式執行時間環境
- MCP 用戶端整合
- 記憶體管理功能
- OpenTelemetry 檢測

ECR 提取快取

解決方案使用 ECR Pull-Through Cache 自動將代理程式映像從公有 ECR 儲存庫分佈到客戶的私有 ECR。此 AWS 受管服務：

- 第一次提取時快取映像 (2-5 分鐘延遲）
- 消除自訂映像複製邏輯
- 提供後續部署的本機映像可用性
- 為每個部署建立唯一的快取規則，以避免衝突

組態儲存

代理程式組態與現有的使用案例組態一起存放在 DynamoDB 中。每個組態包括：

- 系統提示範本
- 模型提供者和模型 ID
- 模型參數 （溫度、max_tokens)
- MCP 伺服器參考和端點
- 記憶體設定 （長期記憶體切換）
- 部署中繼資料

映像版本登錄檔

DynamoDB 資料表會追蹤可用的代理程式映像版本及其快取 URIs，啟用版本管理和回溯相容性。

代理程式組態

系統提示

系統提示會定義客服人員行為、人格和功能。管理員使用者可以：

- 透過客服人員建置器 UI 編輯預設範本
- 包含工具用量和回應格式的說明
- 隨時重設為預設範本

模型選取

代理程式建置器支援 v4.0.0 中的 Amazon Bedrock 模型：

- 模型提供者：Amazon Bedrock （僅限 v4.0.0 中的選項）
- 模型選擇：Claude、Nova 和其他 Bedrock 模型
- 模型參數：溫度、max_tokens、top_p 和模型特定設定

MCP 伺服器整合

模型內容通訊協定伺服器可讓代理程式存取企業工具和資料：

- 透過 GET /mcp API 端點的伺服器探索
- 動態組態，無需變更程式碼
- 身分驗證和端點管理
- 客服人員的工具功能公開

串流和處理

即時串流

Agent Builder 使用從 AgentCore 橋接到 WebSocket 的 Server-Sent Events (SSE) 進行即時回應串流：

- Lambda 函數會建立與 AgentCore 執行期的 SSE 連線
- 串流會橋接至 API Gateway WebSocket
- 啟用token-by-token回應交付至用戶端
- 維持長時間執行請求的連線

處理限制條件

v4.0.0 中的代理程式處理僅限於 Lambda 執行逾時：

- 最長處理時間：15 分鐘
- 同步處理模型
- 適合對話客服人員和中等工作流程
- 針對 v4.1+ 規劃的延伸非同步支援

記憶體管理

短期記憶體

使用自訂 MemoryHookProvider 的所有客服人員預設啟用：

- 透過 Strands 回呼處理常式擷取對話事件
- 依 actorId 和 sessionId 組織內容隔離
- 在工作階段中維護對話內容
- 自動整合 AgentCore 記憶體

長期記憶體

從 strands_tools 使用 AgentCore Memory Tool 的選用功能：

- 客服人員建置器使用者介面中的簡單切換
- 具有預設設定的語意記憶體策略
- 透過自然工具呼叫的代理程式控制存取
- 跨工作階段存放擷取的洞見
- 使用 conversationId 做為 sessionId

可觀測性

AWS OpenTelemetry Distro (ADOT)

容器建置期間會自動檢測代理程式：

- 自動產生代理程式操作的追蹤
- 跨服務界限的分散式追蹤
- 具有相互關聯 IDs結構式記錄

- 與 CloudWatch 交易搜尋整合

身分驗證流程

使用者透過 Amazon Cognito 使用自訂 Lambda 授權方驗證的 JWT 權杖進行身分驗證，該授權方會根據使用者群組從 DynamoDB 擷取 IAM 政策。

工作流程建置器

工作流程建置器透過建立主管客服人員，使用客服人員做為工具委派模式來協調多個客服人員建置器客服人員，以啟用多重客服人員協調。

工作流程架構

關鍵元件

- 主管客服人員：接收使用者請求並委派給專業客服人員的進入點客服人員
- 專門客服人員：客服人員建置器使用案例已註冊為主管的工具
- 代理程式登錄檔：存放代理程式組態和中繼資料的 DynamoDB 資料表
- 協同運作層：將 代理程式的 SDK 實作作為工具模式

客服人員執行個體化

建立本機代理程式

所有專門代理程式都會在相同的 AgentCore 執行時間內於本機執行個體化：

1. 從 DynamoDB 擷取代理程式組態
2. 建立每個 Agent Builder 代理程式的本機執行個體
3. 每個代理程式都會維護自己的 MCP 伺服器連線
4. 主管客服人員將專業客服人員註冊為工具
5. Strands SDK 管理客服人員選擇和委派

規劃您的部署

本節說明規劃部署的成本、安全性、區域和配額考量。

> **Important**
>
> 此解決方案利用 Amazon Bedrock 作為存取 AI 產生模型的主要服務。您必須先請求存取模型，才能在解決方案中使用模型。如需詳細資訊，請參閱《Amazon Bedrock 使用者指南》中的模型存取。

支援的 AWS 區域

> **Important**
>
> 此解決方案可選擇性地使用 Amazon Bedrock 和 Amazon Kendra 服務，但目前尚未在所有 AWS 區域提供。您必須在提供這些服務的 AWS 區域中啟動此解決方案。如需各區域 AWS 服務的最新可用性，請參閱 AWS 區域服務清單。

下列 AWS 區域支援 AWS 上的生成式 AI 應用程式建置器：

區域名稱	
美國東部 (俄亥俄)	加拿大 (中部)
美國東部 (維吉尼亞北部)	歐洲 (法蘭克福)
美國西部 (加利佛尼亞北部)	歐洲 (愛爾蘭)
美國西部 (奧勒岡)	歐洲 (倫敦)
亞太地區 (孟買)	歐洲 (米蘭)
亞太地區 (首爾)	Europe (Paris)
亞太地區 (新加坡)	歐洲 (斯德哥爾摩)

區域名稱	
亞太地區 (悉尼)	Middle East (Bahrain)
亞太地區 (東京)	南美洲 (聖保羅)

> **Note**
>
> 如果在部署中使用在 AWS 外部存取的基礎模型，請洽詢模型提供者其 APIs 可使用的區域。如果其 APIs僅適用於某些區域，您可能會遇到高延遲或甚至逾時形式的不穩定。與您組織的法務和合規團隊進行檢查，以評估資料跨區域界限的考量也很重要。

Cost

使用此 AWS 解決方案，您只需為使用的資源付費，而且沒有最低費用或設定費用。使用者支付用於啟動生成式 AI 使用案例的儀表板，以及部署的任何使用案例的費用。部署的使用案例成本取決於組態。範例組態：

1. 簡單的部署儀表板，每月大約 20 USD。
2. 一種簡單的生產就緒聊天機器人使用案例，使用執行於美國東部（維吉尼亞北部）的預設設定進行部署，由 Amazon Bedrock 提供支援，無法存取文件，每月大約 200 USD。
3. Amazon VPC 使用案例中的擴展系統，每天支援 8，000 個查詢，超過數萬份文件，每月約 1，500 USD。使用案例的成本會根據組態而有所不同，例如不同模型提供者的文字使用案例、是否啟用擷取增強生成 (RAG)，以此類推。

工作負載說明	預估成本（美元/月）
部署儀表板的範例成本	每月 20 美元
文字型概念驗證的範例成本 （包括部署儀表板和 1 個文字使用案例，每天約 100 個互動）	每月 40 美元
高可擴展性生成式 AI 查詢引擎的範例成本	每月 1，500 美元

工作負載說明	預估成本（美元/月）
（包括部署儀表板、1 個文字使用案例，以及 RAG 的 Amazon Kendra 索引，每天最多 100K 份具有 ~8K 個查詢的文件，且已啟用 VPC	
代理程式型概念驗證的範例成本 （包括部署儀表板、1 個已啟用 Amazon Bedrock 知識庫和 Amazon Bedrock 護欄的 Bedrock 代理程式使用案例、每天約 100 個互動）	每月 840 美元
MCP 伺服器的範例成本 （包括部署儀表板、1 個具有 Lambda 整合閘道方法的 MCP 伺服器使用案例、每天約 100 個工具調用）	每月 22 美元
客服人員建置器的範例成本 （包括部署儀表板、1 個啟用 MCP 整合和長期記憶體的客服人員建置器使用案例、每天約 100 個互動）	每月 55 美元
工作流程建置器的範例成本 （包括部署儀表板、具有 3 個客服人員建置器代理程式的 1 個工作流程、每天約 100 個互動）	每月 109 美元

Important

這些範例僅用於協助您預估特定工作負載的成本。使用不同的 LLMs、組態或 AWS 服務可以變更您的成本（例如，無伺服器/隨需計費與佈建/定時計費）。若要管理成本，建議您透過 AWS Cost Explorer 建立預算。價格可能變動。如需完整詳細資訊，請參閱此解決方案中使用的每個 AWS 服務的定價網頁。

執行部署儀表板的範例成本

下表提供一個月美國東部（維吉尼亞北部）區域中具有預設參數和 100 個作用中使用者之部署儀表板的成本明細，約每月 20 美元。

AWS 服務	維度	成本 【美元】
API Gateway、DynamoDB、CloudFront、Amazon S3、Lambda、Systems Manager 參數存放區	每月 5，000 個 512 KB REST API 呼叫，未啟用快取	1.97 美元
Amazon Cognito	每月 100 個啟用進階安全功能且沒有透過 SAML 或 OIDC 聯合身分登入的使用者	5.55 美元
AWS WAF	跨 1 個 Web ACL 和 7 個定義規則的 10，000 個 Web 請求，而沒有任何規則群組	12.60 美元
部署儀表板總成本		20.12 美元

文字型概念驗證的範例成本

部署儀表板可以在指定時間部署許多使用案例。下表顯示使用 LLM 每天執行 100 個查詢的 1 個商業使用者在沒有 RAG 的情況下部署的使用案例的成本明細。查詢會以 WebSocket 上的文字訊息傳送，回應會以字符的形式串流回，並假設已啟用串流。使用 Amazon Bedrock Nova Pro 模型，執行此使用案例的成本約為每月 20 美元。

AWS 服務	維度	成本 【美元】
API Gateway (WebSocket)、CloudFront、Lambda、Amazon S3、AWS Systems Manager 參數存放區	每天 100 次聊天互動。每個訊息的平均訊息大小為 32 KB，每個連線為 5 分鐘。	0.61 美元

AWS 服務	維度	成本 【美元】
CloudWatch	在 上使用詳細模式的 1.5 GB CloudWatch 日誌進行實驗	7.23 美元
Amazon DynamoDB	對話歷史記錄表、1 GB 儲存體 LLM 組態資料表、1 GB 儲存體	3.05 美元
使用案例成本的小計 （不包括 LLMs)		10.89 美元
Amazon Bedrock (Nova Pro)	每天 100 次互動的假設： * 每天 190K個輸入字符的每月成本 = $0.152 × 30 * 每天 16K個輸出字符的每月成本 = $0.0512 × 30	6.10 美元
Amazon Bedrock (Nova Pro) 的應用程式總成本	10.89 美元 （使用案例成本） + 6.10 美元 (Amazon Bedrock 成本）	17.00 美元

> **Note**
>
> 這些預估值不包含對 AWS 網路外部服務進行的推論呼叫成本。如果您未使用 AWS 模型提供者，請參閱 LLM 提供者的定價指南。
> 如需 AWS 服務的定價指南，請參閱：Amazon Bedrock 定價和 Amazon SageMaker AI 定價。

高可擴展性生成式 AI 查詢引擎的範例成本

下表提供啟用 RAG 的使用案例的成本明細，並以 Amazon Bedrock 的 Nova Pro 模型做為 LLM。新增 Bedrock 知識庫時，此使用案例的費用約為每月 1300 美元

AWS 服務	維度	成本 【美元】
API Gateway (WebSocket)	每天 8000 次聊天互動。每個訊息的平均訊息大小為 32 KB，每個連線為 5 分鐘。	38.89 美元
CloudFront	每月 240，000 個請求，100 GB 資料傳輸到網際網路，1 GB 資料傳輸到原始伺服器	8.76 美元
Amazon Bedrock (Nova Pro)	假設： 輸入字符 = promptTemplate (400) + 內容 (400)+ chatHistory (1080) + 查詢 輸入字符 (20)= 1，900 輸出字符 = 160 （平均） 每天有 8，000 筆交易， 每日輸入字符成本 (1，900 x 8，000 = 15，200，000 個字符 x 每個字符 0.0008/1000 價格） 每日輸出權杖成本 (160 x 8，000 = 1，280，000 個權杖 x 每個權杖 0.0032/1000 價格） 每月成本 (($12.16 + $4.10) x 30)	487.80 美元
CloudWatch	使用 5 GB 資料擷取日誌和 1 個儀表板的 24 個指標	9.72 美元
DynamoDB	DynamoDB 資料表可追蹤每個記錄的對話歷史記錄，每天最	11.70 美元

AWS 服務	維度	成本 【美元】
	多 1 KB 資料、8，000 次讀取和寫入	
Lambda	容器大小 - 128 MB、512 MB 暫時性 儲存，2 個用於授權的 Lambda 函數 容器大小 - 256 MB、512 MB 暫時性儲存、每秒 5 個請求與 20 秒平均運算時間	20.89 美元
總使用案例成本		每月 577.76 美元 + 知識庫成本 （ 請參閱下文 ）

> **Note**
>
> 對 AWS 網路外的任何服務發出的 API 呼叫成本不包含在這些預估值中。如果未使用 Amazon Bedrock，請參閱 LLM 供應商的定價指南。

新增知識庫的成本

知識庫成本會根據使用的知識庫類型，以及 （ 在 Bedrock 的情況下 ） 知識庫使用的後端向量存放區而有所不同。佈建和管理知識庫不在解決方案的範圍內。

Amazon Bedrock 知識庫

解決方案不會管理或佈建任何與 Amazon Bedrock 知識庫相關的資源。Amazon Bedrock 使用知識庫功能本身不會產生費用，但每次查詢使用案例所使用的內嵌模型，都會向您收取費用。此外，您知識庫的後端向量存放區 （ 例如，Amazon OpenSearch Service 中的索引，或 Amazon Relational Database Service 內的資料庫 ） 將產生無法在此處提供或計算的相關費用。

對於上述高度可擴展的生成式 AI 查詢引擎案例，此服務呼叫 Amazon Bedrock 內嵌模型所產生的成本如下：

AWS 服務	維度	成本 【美元】
Amazon Bedrock (Amazon Titan Text Embeddings V2)	每天 8，000 個查詢，每個查詢 1，900 個輸入字符 = 15，200，000 個字符 = 每天 0.30 USD。 每日成本 x 30 天 = 每月成本 9.00 USD	9.00 美元
Amazon OpenSearch Service (Serverless) 範例用量	使用 4 x OpenSearch 運算單位 (OCU) 的基本無伺服器組態（最低計費）= 每天 23.04 USD 每日費用 x 30 天 = 691.20 USD **Note** 這可提供粗略的預估，因為某些工作負載將需要更多 OCUs，而具有現有佈建 OpenSearch 資源的客戶將在此產生較低的成本。	691.20 美元
總額外成本		700.20 美元

Amazon Kendra

解決方案可以為您佈建 Kendra 索引，也可以自備。執行適用於上述高可擴展性生成式 AI 查詢引擎的組態的成本如下：

AWS 服務	維度	成本 【美元】
Amazon Kendra	每天 0-8，000 個查詢，以及最多 100，000 個具有 0-50 個資料來源的 Amazon Kendra Enterprise Edition 文件	1，008.00 美元

> **Note**
>
> 您可以在使用案例之間共用 Amazon Kendra 索引，但這可能會增加每個索引的查詢數量。如果這不屬於 Amazon Kendra Enterprise Edition，則會收取額外費用。

為使用案例啟用 Amazon VPC 的成本增加

下表提供在兩個 AZs 中部署的使用案例啟用 Amazon VPC 的成本明細。

AWS 服務	維度	成本 【美元】
Amazon NAT 閘道	假設：2 個可用區域部署，每個可用區域都有 NAT Gateway。透過 NAT Gateway 處理的 100 GB 資料 730 小時，每月處理 100 GB 資料	74.70 美元
AWS PrivateLink (VPC 端點)	假設：2 個 AZ 部署，每個 AZ 具有 1 個私有子網路，1 個 VPC 端點具有 2 個彈性網路介面 (ENIs)。 6 個 VPC 端點，每個 VPC 端點 2 ENIs，730 小時，每月處理 1，024 GB 的資料	97.84 美元
公有 IPv4 地址	假設：2 個可用區域部署，每個可用區域 1 個公有子網路，每個公有子網路都有 NAT	7.30 美元

AWS 服務	維度	成本 【美元】
	Gateway。每個使用 1 個作用中公有 IPv4 設定的 NAT 閘道。 2 個作用中公有 IPv4 地址 x 每月 730 小時 x 0.005 美元每小時費用 = 7.3 美元	
額外費用 （適用於 Amazon VPC)		179.93 美元

使用佈建輸送量的成本影響

佈建輸送量成本會根據您已佈建的模型類型和承諾期間，以及承諾期間選取的模型單位而有所不同。使用佈建輸送量會產生額外費用。

如需詳細資訊和up-to-date定價，請參閱 Bedrock 定價。

使用跨區域推論的成本

使用跨區域推論的路由或資料傳輸無需額外費用。您為模型支付的價格與來源或主要區域中的模型相同。

代理程式型概念驗證的範例成本

當您使用 Amazon Bedrock 代理程式時，系統會根據包含代理程式的元件向您收費，例如後端模型和知識庫 （如果已啟用 RAG)，以及您新增的其他功能。下表顯示使用隨需 Claude 3.5 Sonnet 模型、Amazon Bedrock 知識庫和 Amazon Bedrock 護欄設定的 Bedrock 代理程式使用案例的成本明細。

與新增 Amazon Bedrock 知識庫的成本類似，此解決方案不會管理或佈建與 Amazon Bedrock 代理程式相關的資源。此解決方案也不會產生使用 Amazon Bedrock 知識庫的成本，但會產生下列成本：

- 針對傳送給它的每個查詢使用內嵌模型
- 知識庫的後端向量存放區 （例如，Amazon OpenSearch Service 中的索引，或 Amazon RDS 內的資料庫）

下表假設每天有 100 個互動，每個查詢有 1，900 個輸入字符和 160 個輸出字符。

> **Note**
>
> 對於此範例 Bedrock 代理程式使用案例，如果有設定為使用外部 API 的動作群組，則這些成本會是額外的。它們超出此資料表中計算的範圍。

AWS 服務	維度	成本 【美元】
API Gateway (WebSocket)、CloudFront、Lambda、Amazon S3、Systems Manager 參數存放區	每天 100 次聊天互動，每則訊息平均大小 32 KB，每條連線 5 分鐘	0.61 美元
CloudWatch	在 上使用詳細模式的 1.5 GB CloudWatch Logs 進行實驗	7.23 美元
DynamoDB	1KB 記錄大小和 1 GB 儲存體的 LLM 組態資料表	0.25 美元
成本小計 （不包括 LLMs)		8.09 美元
Anthropic Claude 3.5 Sonnet	* 每天 190K個輸入字符的每日成本 (0.003/1，000 個字符） = 0.57 美元 + 每日成本 × 30 天 = 17.10 美元 * 每天 16K 個輸出字符的每日成本 (0.015/1，000 個字符） = 0.24 美元 + 每日成本 × 30 天 = 7.20 美元	24.30 美元
適用於 Amazon Bedrock 知識庫的 Amazon Bedrock (Amazon Titan Text Embeddings V2)	每天 190K個輸入字符的每日成本 (0.00002/1000 字符） = 0.004 每日成本 × 30 天 = 0.12 美元	0.12 美元

AWS 服務	維度	成本 【美元】
Amazon OpenSearch Service (Serverless) 範例用量	使用 4 × OpenSearch 運算單位 (OCU) 的基本無伺服器組態（最低計費） = 每天 23.04 美元 每日成本 × 30 天 = 691.20 美元	691.20 美元
Amazon Bedrock 防護機制	190K 權杖大約等同於 760K (190，000 × 4) 字元和 3，800 個文字單位 (760K 字元/200) 考慮使用內容篩選條件、個人身分識別資訊 (PII) 篩選條件、敏感資訊篩選條件（規則表達式）和單字篩選條件設定的護欄 每日內容篩選條件成本 (0.75/1000 文字單位）+ PII 篩選條件成本 (0.1/1000 文字單位）+ 敏感資訊篩選條件 (regex) + 文字篩選條件 = $2.85 + $0.38 + $0 + $0 每月成本 = 每日成本 × 30 天 = 96.90 美元	96.90 美元
Anthropic Claude 3.5 Sonnet 支援的代理程式應用程式總成本	$8.09（使用案例成本）+ $812.52（其他代理程式組態）	820.61 美元

> **Note**
>
> 如果您未使用 AWS 模型提供者，請參閱 LLM 提供者的定價指南。如需 AWS 服務的定價指南，請參閱：Amazon Bedrock 定價和 Amazon SageMaker AI 定價。

MCP 伺服器的範例成本

MCP 伺服器使用案例可在 Amazon Bedrock AgentCore 上部署和管理模型內容通訊協定伺服器。下表顯示使用 Gateway 方法來包裝現有 Lambda 函數的 MCP Server 使用案例的成本明細。

解決方案會管理 AgentCore Gateway 部署和組態。您需要支付以下費用：

- 基礎設施成本 (API Gateway、Lambda、DynamoDB、CloudWatch、S3)
- AgentCore Gateway 使用量 （每個工具調用）
- Lambda 函數執行成本 （適用於具有 Lambda 目標的閘道方法）
- 外部 API 成本 （適用於具有 API 或 MCP 伺服器目標的閘道方法，如適用）

項目	計算	Cost
Amazon API Gateway (REST API)	每天 100 個工具呼叫 × 30 天 = 每月 3，000 個請求	0.05 USD
AWS Lambda （協同運作）	每天 100 次叫用 × 30 天 × 1 秒平均 × 512 MB = 每月 3，000 GB-秒	0.05 USD
Amazon DynamoDB	每月 3，000 個讀取/寫入請求 + 1 GB 儲存	0.15 美元
Amazon CloudWatch	3，000 個調用的標準監控和記錄	1.00 美元
Amazon S3	組態儲存和日誌 （最低用量）	0.25 美元
Amazon Bedrock AgentCore Gateway	每月 3，000 個工具叫用	0.05 USD

項目	計算	Cost
目標 Lambda 函數	每天 100 次叫用 × 30 天 × 0.5 秒 × 128 MB = 每月 1，500 GB 秒	0.25 美元
每月成本總計	1.75 美元（基礎設施）+ 0.05 美元 (AgentCore Gateway)	1.80 美元

> **Note**
>
> 成本會根據部署方法（閘道與執行時間）、目標類型和用量模式而有所不同。執行期方法部署會產生 AgentCore 執行期費用，而不是 Gateway 費用。外部 API 成本和自訂容器託管成本是額外的。

客服人員建置器的範例成本

Agent Builder 可讓您在 Amazon Bedrock AgentCore 上建立和部署自訂代理程式。下表顯示使用 Claude 3.5 Sonnet、MCP 伺服器整合和啟用長期記憶體設定的 Agent Builder 使用案例的成本明細。

解決方案會管理 AgentCore 執行期部署和組態。您需要支付以下費用：

- 基礎設施成本 (API Gateway、Lambda、DynamoDB、CloudWatch、S3)
- AgentCore 執行期耗用（根據實際代理程式執行時間的 CPU 和記憶體時數）
- 基礎模型推論（輸入和輸出字符）
- AgentCore 記憶體（短期事件和長期儲存/擷取）

下表假設每天有 100 個互動，每個查詢有 1，900 個輸入字符和 160 個輸出字符，每個互動的平均客服人員執行時間為 5 秒。

AWS 服務	維度	成本 【美元】
API Gateway (WebSocket)、CloudFront、Lamb	每天 100 次聊天互動，平均訊息大小每則訊息 32 KB，每則連線 5 分鐘	0.61 美元

AWS 服務	維度	成本 【美元】
da、Amazon S3、Systems Manager 參數存放區		
CloudWatch	在 上使用詳細模式的 1.5 GB CloudWatch Logs 進行實驗	7.23 美元
DynamoDB	1KB 記錄大小和 1 GB 儲存體的 LLM 組態資料表	0.25 美元
基礎設施成本的小計		8.09 美元
Amazon Bedrock AgentCore 執行期	* CPU：1 個 vCPU × 5 秒 × 100 個互動 = 125 個 vCPU-秒/天 = 0.140 個 vCPU-小時/天 + 每日成本：0.140 × $0.0895 = $0.013 + 每月成本：$0.013 × 30 = $0.38 * 記憶體：512 MB (0.5 GB) × 5 秒 × 100 次互動 = 250 GB-秒/天 = 0.069 GB-小時/天 + 每日成本：0.069 × $0.00945 = $0.0007 + 每月成本：$0.0007 × 30 = $0.02	0.40 美元
Anthropic Claude 3.5 Sonnet	* 每天 190K個輸入字符的每日成本 (0.003/1，000 個字符） = $0.57 + 每日成本 × 30 天 = $17.10 * 每天 16K000 個輸出字符的每日成本 (0.015/1，000 個字符） = $0.24 + 每日成本 × 30 天 = $7.20	24.30 美元

AWS 服務	維度	成本 【美元】
Amazon Bedrock AgentCore 記憶體	* 短期記憶體：100 個新事件/天 × 0.25/1，000 美元事件 = 0.025 美元/天 + 每月成本：0.025 美元 × 30 = 0.75 美元 * 長期記憶體儲存（內建策略）：100 筆記錄 × $0.75/1，000 筆記錄/月 = $0.075/月 * 長期記憶體擷取：100 次擷取/天 × 0.50 美元/1，000 次擷取 = 0.05 美元/天 + 每月成本：0.05 美元 × 30 = 1.50 美元	2.33 美元
搭配 Claude 3.5 Sonnet 的 Agent Builder 應用程式總成本	$8.09（基礎設施）+ $0.40 (AgentCore 執行期）+ $24.30（模型）+ $2.33（記憶體）	35.12 美元

Note

AgentCore 執行期定價以耗用量為基礎。實際成本取決於：

- 代理程式執行時間（作用中處理期間的 CPU 和記憶體用量）
- 互動次數及其複雜性
- MCP 工具用量（用於工具執行的額外 CPU/記憶體）
- 記憶體組態（啟用短期與長期記憶體）

如需 AgentCore 定價的詳細資訊，請參閱 Amazon Bedrock 定價。

> **Note**
>
> 如果使用叫用外部 APIs或服務的 MCP 伺服器，則這些成本是額外成本，且超出此計算的範圍。同樣地，如果使用 AgentCore 瀏覽器或 Code Interpreter 工具，每 vCPU 小時收費 0.0895 美元，每 GB 小時收費 0.00945 美元。

工作流程建置器的範例成本

工作流程建置器會建立協調多個客服人員建置器客服人員的主管客服人員。下表顯示具有 1 個主管代理程式和 3 個專業代理程式建置器代理程式的工作流程的成本明細，所有設定都已啟用 Claude 3.5 Sonnet 和長期記憶體。

假設：每天 100 次互動，每次互動平均 2 次客服人員委派，每個客服人員 5 秒執行時間。

AWS 服務	維度	成本 【美元】
API Gateway (WebSocket)、CloudFront、Lambda、Amazon S3、Systems Manager 參數存放區	每天 100 次聊天互動，平均訊息大小每則訊息 32 KB，每則連線 5 分鐘	0.61 美元
CloudWatch	在 上使用詳細模式的 1.5 GB CloudWatch Logs 進行實驗	7.23 美元
DynamoDB	1KB 記錄大小和 1 GB 儲存體的 LLM 組態資料表	0.25 美元
基礎設施成本的小計		8.09 美元
Amazon Bedrock AgentCore 執行期 （主管代理程式）	* CPU：1 個 vCPU × 5 秒 × 100 個互動 = 0.140 個 vCPU 小時/天 × 30 = $0.38 * 記憶體：0.5 GB × 5 秒 × 100 個互動 = 0.069 GB 小時/天 × 30 = $0.02	0.40 美元
Amazon Bedrock AgentCore 執行期 (3 個專用代理程式）	* 平均每次互動 2 次委派 = 200 次客服人員執行/天 * CPU：1	0.79 美元

AWS 服務	維度	成本 【美元】
	次 vCPU × 5 秒 × 200 = 0.278 次 vCPU-小時/天 × 30 = $0.75 * 記憶體：0.5 GB × 5 秒 × 200 = 0.139 GB-小時/天 × 30 = $0.04	
Anthropic Claude 3.5 Sonnet （主管代理程式）	* 輸入：190K 權杖/天 × $0.003/1K = $0.57/天 × 30 = $17.10 * 輸出：16K 權杖/天 × $0.015/1K = $0.24/天 × 30 = $7.20	24.30 美元
Anthropic Claude 3.5 Sonnet （專用代理程式）	* 每次互動平均 2 次委派 * 輸入：380K 權杖/天 × 0.003/1K = 1.14 美元/天 × 30 = 34.20 美元 * 輸出：32K 權杖/天 × 0.015/1K = 0.48 美元/天 × 30 = 14.40 美元	48.60 美元
Amazon Bedrock AgentCore 記憶體 （主管代理程式）	* 短期：100 個事件/天 × $0.25/1K × 30 = $0.75 * 長期儲存：100 個記錄 × $0.75/1K = $0.08 * 長期擷取：100 個擷取/天 × $0.50/1K × 30 = $1.50	2.33 美元
Amazon Bedrock AgentCore 記憶體 （專用代理程式）	* 短期：200 個事件/天 × $0.25/1K × 30 = $1.50 * 長期儲存：200 個記錄 × $0.75/1K = $0.15 * 長期擷取：200 個擷取/天 × $0.50/1K × 30 = $3.00	4.65 美元
具有 3 個代理程式之工作流程建置器的應用程式總成本	$8.09 （基礎設施） + $1.19 (AgentCore 執行期） + $72.90 （模型） + $6.98 （記憶體）	89.16 美元

Note

- 較高的委派率會按比例增加字符消耗

如需 AgentCore 定價的詳細資訊，請參閱 Amazon Bedrock 定價。

安全

當您在 AWS 基礎設施上建置系統時，安全責任將由您與 AWS 共同承擔。此共同責任模型可減輕您的營運負擔，因為 AWS 會操作、管理和控制元件，包括主機作業系統、虛擬化層，以及服務營運所在設施的實體安全性。如需 AWS 安全性的詳細資訊，請造訪 AWS Cloud Security。

在 Amazon Bedrock 上使用基礎模型

Amazon Bedrock 託管從 Amazon Nova 模型到其他主要基礎模型 (FMs) 的一系列模型。使用 Amazon Bedrock 時，所有模型都會託管在 AWS 基礎設施中。這表示使用 Amazon Bedrock 做為 LLM 提供者時，所有推論請求都會保留在 AWS 網路中，網路流量也不會離開您的區域。

Note

透過 Amazon Bedrock 提供的所有基礎模型 (FMs) 都直接託管在 AWS 管理和擁有的 AWS 基礎設施上。模型提供者無法存取客戶資料，例如提示和接續，或 Amazon Bedrock 服務日誌。如需有關 Amazon Bedrock 安全狀態的其他資訊，請參閱《Amazon Bedrock 使用者指南》中的 Amazon Bedrock 中的資料保護。

IAM 角色

IAM 角色可讓客戶將精細存取政策和許可指派給 AWS 雲端上的服務和使用者。此解決方案會建立 IAM 角色，授予解決方案的 Lambda 函數建立區域資源的存取權。

CloudWatch Logs

您可以在使用部署儀表板模型選擇頁面，在其他設定下部署使用案例時啟用詳細模式。詳細模式可啟用詳細的 CloudWatch 日誌，有助於偵錯和快速實驗。

> **Note**
>
> 啟用詳細模式時，也會記錄從知識庫擷取的文件（如果啟用 RAG) 和提示，其中可能包含敏感資訊。

VPC

解決方案提供兩種 Amazon VPC 組態選項：

1. 讓解決方案為您建置 Amazon VPC。
2. 管理和攜帶您自己的 Amazon VPC 以在解決方案中使用。

讓解決方案為您建置 Amazon VPC

如果您選擇讓解決方案建置 Amazon VPC 的選項，預設會部署為 2-AZ 架構，CIDR 範圍為 10.10.0.0/20。您可以選擇使用 Amazon VPC IP Address Manager (IPAM)，在每個 AZ 中具有 1 個公有子網路和 1 個私有子網路。解決方案會在每個公有子網路中建立 NAT Gateway，並設定 Lambda 函數在私有子網路中建立 ENIs。此外，此組態會建立路由表及其項目、安全群組及其規則、網路 ACLs、VPC 端點（閘道和介面端點）。

管理您自己的 Amazon VPC

使用 Amazon VPC 部署解決方案時，您可以選擇在 AWS 帳戶和區域中使用現有的 Amazon VPC。我們建議您在至少兩個可用區域中提供 VPC，以確保高可用性。您的 VPC 也必須具有下列 VPC 端點，以及 VPC 和路由表組態的相關 IAM 政策。

對於部署儀表板 Amazon VPC

1. DynamoDB 的閘道端點。
2. S3 的閘道端點。
3. CloudWatch 的界面端點。
4. AWS CloudFormation 的界面端點。

對於使用案例 Amazon VPC

1. DynamoDB 的閘道端點。

2. S3 的閘道端點。
3. CloudWatch 的界面端點。
4. Systems Manager 參數存放區的界面端點。

> **Note**
>
> 解決方案只需要 `com.amazonaws.region.ssm`。

5. Amazon Bedrock 的界面端點 (bedrock-runtime、agent-runtime、bedrock-agent-runtime)。
6. 選用：如果部署將使用 Amazon Kendra 作為知識庫，則需要 Amazon Kendra 的介面端點。
7. 選用：如果部署將使用 Amazon Bedrock 下的任何 LLM，則需要 Amazon Bedrock 的介面端點。

> **Note**
>
> 解決方案只需要 `com.amazonaws.region.bedrock-runtime`。

8. 選用：如果部署將針對 LLM 使用 Amazon SageMaker AI，則需要 Amazon SageMaker AI 的介面端點。

> **Note**
>
> 使用自備 VPC 部署選項時，解決方案不會刪除或修改 VPC 組態。不過，它會刪除解決方案在為我建立 VPCs選項中建立的任何 VPC。 因此，在堆疊/部署之間共用解決方案管理的 VPC 時，您必須小心。
> 例如，部署 A 使用為我建立 VPC 選項。部署 B 使用使用部署 A 建立的 VPC 自帶 VPC。如果部署 A 在部署 B 之前遭到刪除，則部署 B 將無法再運作，因為已刪除 VPC。此外，由於部署 B 使用 Lambda 函數建立的 ENIs，刪除部署 A 可能會發生錯誤並保留剩餘資源。

Amazon CloudFront

此解決方案會部署託管在 Amazon S3 儲存貯體中的 Web 主控台。為了協助減少延遲並改善安全性，此解決方案包含具有原始存取身分的 CloudFront 分佈，這是提供對解決方案網站儲存貯體內容公開存取的 CloudFront 使用者。如需詳細資訊，請參閱《Amazon CloudFront 開發人員指南》中的使用原始存取身分限制對 Amazon S3 內容的存取。

> **Note**
>
> CloudFront 的帳戶層級軟配額限制為 20 個回應標頭政策。基於安全考量，此解決方案會建立自訂回應標頭政策。如果您在 AWS 或其使用案例上有 20 個以上的生成式 AI 應用程式建置器部署，新部署可能會因為達到配額限制而失敗。

若要解決此問題，您可以依照下列步驟，在 AWS Service Quotas 主控台中請求增加回應標頭政策配額：

1. 開啟 AWS Service Quotas 主控台。
2. 在導覽窗格中，選擇 AWS services (AWS 服務)。
3. 搜尋並選取 Amazon CloudFront。
4. 捲動至回應標頭政策配額，然後選擇請求配額增加。
5. 按照提示請求提高 AWS 帳戶的配額限制。

透過增加回應標頭政策配額，您可以確保 AWS 上的生成式 AI 應用程式建置器或其使用案例的新部署不會因為配額限制而失敗。

配額

服務配額 (也稱為限制) 是您 AWS 帳戶的服務資源或操作數目最大值。

此解決方案中 AWS 服務的配額

請確定您為此解決方案中實作的每個服務有足夠的配額。如需詳細資訊，請參閱 AWS 服務配額。

使用以下連結前往該服務的 頁面。若要在不切換頁面的情況下檢視文件中所有 AWS 服務的服務配額，請改為檢視 PDF 中服務端點和配額頁面中的資訊。

Amazon Bedrock AgentCore 配額

對於 Agent Builder 部署，請注意下列 Amazon Bedrock AgentCore 服務配額：

配額	美國東部 (維吉尼亞北部)	其他區域
每個帳戶的作用中工作階段工作負載	1000	500
每個帳戶的客服人員總數	1,000	1,000
每個帳戶的版本	1,000	1,000

部署解決方案

此解決方案使用 AWS CloudFormation 範本和堆疊來自動化其部署。CloudFormation 範本會指定此解決方案中包含的 AWS 資源及其屬性。CloudFormation 堆疊會佈建範本中所述的資源。

部署程序概觀

啟動解決方案之前，請檢閱本指南中討論的成本、架構、安全性和其他考量事項。

> **Important**
>
> 如果您計劃使用 Amazon Bedrock，您必須在模型可供使用之前請求存取模型。如需詳細資訊，請參閱《Amazon Bedrock 使用者指南》中的模型存取。

部署時間：約 10 分鐘

步驟 1：啟動部署儀表板堆疊

步驟 2：部署使用案例

步驟 3：使用部署儀表板精靈部署使用案例

步驟 4：部署後組態

或者，如果您不想使用部署儀表板 UI 或 APIs，則可以與解決方案分開部署使用案例。

- 部署獨立文字使用案例
- 部署獨立的 Bedrock 代理程式使用案例

您也可以提供 DynamoDB 聊天組態。

> **Important**
>
> 此解決方案會將有關使用此解決方案的操作指標傳送給 AWS（「資料」）。我們使用此資料來更好地了解客戶如何使用此解決方案和相關的服務和產品。AWS 收集此資料受 AWS 隱私權政策約束。

AWS CloudFormation 範本

您可以在部署之前下載此解決方案的 CloudFormation 範本。

View template

generative-ai-application-builder-on-aws.template - 使用此範本啟動解決方案和所有相關元件。預設組態會部署在此解決方案區段的 AWS 服務中找到的核心和支援解決方案，但您可以自訂範本以符合您的特定需求。

> **Note**
>
> AWS CloudFormation 資源是從 AWS 雲端開發套件 (AWS CDK) 建構模組建立。

此 AWS CloudFormation 範本會在 AWS 雲端的 AWS 上部署生成式 AI 應用程式建置器。

步驟 1：啟動部署儀表板堆疊

遵循本節中的step-by-step說明，設定解決方案並將其部署到您的帳戶。

部署時間：約 10 分鐘

1. 登入 AWS 管理主控台，然後選取按鈕以啟動 `generative-ai-application-builder-on-aws.template` CloudFormation 範本。

   Launch solution

2. 根據預設，範本會在美國東部（維吉尼亞北部）區域啟動。若要在不同 AWS 區域中啟動解決方案，請使用主控台導覽列中的區域選擇器。

   > **Note**
   >
   > 此解決方案使用 Amazon Kendra 和 Amazon Bedrock，目前尚未在所有 AWS 區域提供。如果使用這些功能，您必須在提供這些服務的 AWS 區域中啟動此解決方案。如需各區域的最新可用性，請參閱 AWS 區域服務清單。

3. 在建立堆疊頁面上，確認正確的範本 URL 位於 Amazon S3 URL 文字方塊中，然後選擇下一步。

4. 在指定堆疊詳細資訊頁面上，為您的解決方案堆疊指派名稱。如需有關命名字元限制的資訊，請參閱《AWS Identity and Access Management 使用者指南》中的 IAM 和 STS 限制。
5. 在參數下，檢閱此解決方案範本的參數，並視需要修改這些參數。此解決方案使用下列預設值。

參數	預設	Description
管理員使用者電子郵件	No	有權存取部署儀表板的管理員使用者電子郵件地址。如果提供，則會建立具有部署和管理使用案例許可的 Amazon Cognito 群組和使用者。您也可以使用 `placeholder@example.com` 來建立群組，但不能使用 使用者。如需設定使用者集區的相關資訊，請參閱手動使用者集區組態。
VpcEnabled	No	部署儀表板是否部署在 VPC 內
CreateNewVpc	No	只有在 VpcEnabled 為 時才可用`Yes`。如果值為 `Yes`，堆疊會建立 VPC，並在建立的 VPC 內部署解決方案。 如果 VpcEnabled 是 `Yes`，而 CreateNewVpc 是 No，則您必須提供現有的 VPC 組態 (ExistingVpcId、ExistingPrivateSubnetIds、ExistingSecurityGroupIds、VpcAzs)。
IPAMPoolId	（選用輸入）	您可以設定 IPAM，並提供建立的 ID 做為輸入，以指派此堆疊部署應使用的 IP 地址範

參數	預設	Description
		圍。如需 IPAM 的詳細資訊，請參閱 IPAM 的運作方式。
DeployUI	Yes	您可以選擇在沒有 Web 使用者介面（以及 Web 部署所需的 AWS 資源）的情況下部署部署儀表板。在這種情況下，解決方案會部署所有基礎設施，包括 REST API 端點。此選項有助於將您自己的 Web 界面與部署儀表板 APIs 整合。
ExistingVpcId	（選用輸入）	只有在您想要在已建立的現有 VPC 中部署解決方案時才需要。
ExistingPrivateSubnetIds	（選用輸入）	只有在您想要在已建立的現有 VPC 中部署解決方案時才需要。Lambda 函數將部署在此子網路中。
ExistingSecurityGroupIds	（選用輸入）	只有在您想要在已建立的現有 VPC 中部署解決方案時才需要。確保安全群組具有傳出 TCP 連線的許可。
VpcAzs	（選用輸入）	只有在您想要在已建立的現有 VPC 中部署解決方案時才需要。
CognitoDomainPrefix	（選用輸入）	只有在您想要在您建立的現有 Amazon Cognito 使用者集區中部署解決方案時才需要。如果您不提供值，解決方案會產生該值。

參數	預設	Description
ExistingCognitoUserPoolId	（選用輸入）	只有在您想要在您建立的現有 Amazon Cognito 使用者集區中部署解決方案時才需要。
ExistingCognitoUserPoolClient	（選用輸入）	只有在您想要在您建立的現有 Amazon Cognito 使用者集區中部署解決方案時才需要。如果您不提供值，解決方案會建立使用者集區用戶端。只有在您提供 ExistingCognitoUserPoolId 值時，才能提供此參數。

6. 選擇下一步。
7. 在 Configure stack options (設定堆疊選項) 頁面，選擇 Next (下一步)。
8. 在檢閱和建立頁面上，檢閱並確認設定。選取確認範本將建立 AWS Identity and Access Management (IAM) 資源的方塊。
9. 選擇提交以部署堆疊。

您可以在狀態欄的 AWS CloudFormation 主控台中檢視堆疊的狀態。您應該會在大約 10 分鐘內收到 CREATE_COMPLETE 狀態。

步驟 2：部署使用案例

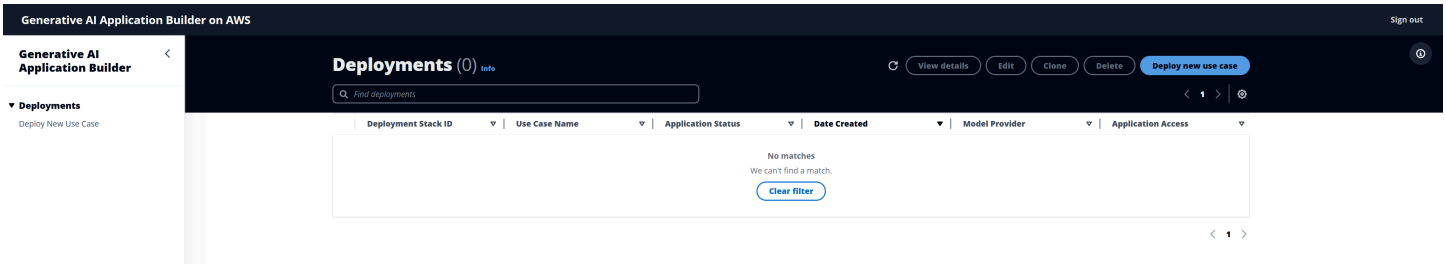
> **Important**
>
> 堆疊成功部署後，註冊電子郵件會傳送至設定的管理員使用者電子郵件。管理員使用者可以使用這些登入資料登入部署儀表板，以使用 Web 應用程式。

> **Note**
>
> 有權存取 AWS 管理主控台的 DevOps 使用者，必須在堆疊完成時為管理員使用者提供部署儀表板 UI 的 CloudFront URL。您可以在 CloudFormation 堆疊的輸出索引標籤中找到 URL。

1. 以管理員使用者身分登入部署儀表板。
2. 在應用程式登陸頁面上，選擇部署新的使用案例。

   這會啟動部署精靈，引導您建置使用案例。

描述部署儀表板登陸頁面 - 全新部署



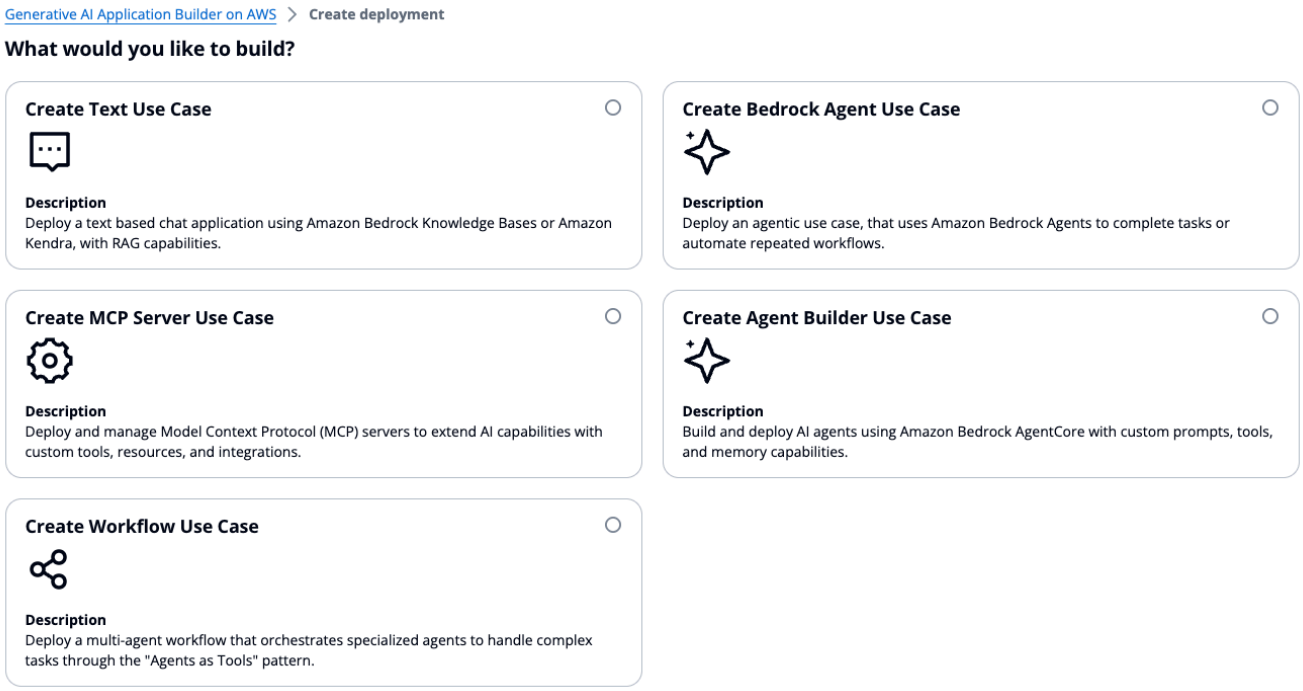
> **Note**
>
> 如果您需要將其他使用者新增至部署，請參閱管理 Cognito 使用者集區以取得更多詳細資訊。

步驟 3：使用部署儀表板精靈部署使用案例

在部署儀表板精靈中，您必須選擇下列項目：

- 文字使用案例 - 部署具有選用 RAG 功能的聊天應用程式
- Bedrock 代理程式使用案例 - 使用 Amazon Bedrock 代理程式來完成任務或自動化重複的工作流程
- MCP 伺服器 - 使用閘道或執行時間方法部署和管理 MCP 伺服器
- Agent Builder - 在 AgentCore 上使用 MCP 整合和記憶體管理建置和部署自訂代理程式
- 工作流程建置器 - 使用階層委派協調多個客服人員建置器代理程式

顯示五個選項：建立文字使用案例、建立 Bedrock 代理程式使用案例、建立 MCP 伺服器使用案例、建立代理程式建置器使用案例或建立工作流程使用案例。



步驟 3a：部署文字使用案例

本節提供部署文字使用案例的說明。

選取使用案例

當您選擇建立文字使用案例時，UI 會開啟選取使用案例畫面。請提供下列資訊：

- 使用案例名稱。
- 要新增至使用案例 Amazon Cognito 使用者集區的使用案例預設使用者選用電子郵件地址，以及獲得與其互動的許可。
- 您是否要使用此使用案例部署 UI。如果您不想使用 使用案例部署 UI，則可以使用已部署的 API 端點來搭配應用程式使用。

使用案例詳細資訊

使用案例詳細資訊步驟可讓您為部署設定其他設定。

根據預設，當解決方案部署部署儀表板時，文字使用案例會為您建立和設定 Amazon Cognito 使用者集區。解決方案會使用相同使用者集區中新建立的用戶端來驗證新的使用案例。不過，如果您想要使用自己的 Amazon Cognito 使用者集區和用戶端搭配使用案例，您可以在此步驟中提供現有的使用者集區 ID 和用戶端 ID。

> **Important**
>
> 透過部署精靈建立 Amazon Cognito 使用者集區時，管理員使用者可以存取所有部署的使用案例。如果您在部署期間提供自己的使用者集區，您必須確保管理員具有存取已部署使用案例的許可。
> 您也需要在 Cognito 的應用程式用戶端中更新允許的回呼 URLs 和允許的登出 URLs。若要執行此作業：
>
> 1. 導覽至 Cognito 主控台
> 2. 選擇 User Pools (使用者集區)。
> 3. 選擇您的使用者集區。
> 4. 選擇左側選單上的應用程式用戶端。
> 5. 選擇您要修改的應用程式用戶端。
> 6. 選擇登入頁面索引標籤。
> 7. 選擇編輯並新增您的 URLs。
> 8. 選擇儲存變更。
>
> 此外，如果您需要將更多使用者新增至使用案例，請參閱管理 Cognito 使用者集區一節。

選取網路組態

此精靈步驟可讓您使用預先存在或新的 Amazon Virtual Private Cloud (Amazon VPC) 部署使用案例。如果選取預先存在的 VPC，您需要提供 VPC ID、最多 16 個子網路 ID 和最多 5 個安全群組 IDs，才能與此 VPC 搭配使用。如果您未使用預先存在的 VPC，則會為您設定這些設定。

選取模型

在選取模型步驟中，您可以從下拉式選單中選擇模型提供者。有兩種選項： Bedrock 和 SageMaker。

如果選取 SageMaker，您可以在 SageMaker AI 主控台中建立 SageMaker AI 模型端點，並提供模型預期的輸入結構描述，以及 LLM 回應的輸出 JSONPath。您可以參考使用 Amazon SageMaker AI 做為 LLM 提供者一節，以及解決方案 GitHub 儲存庫中提供的 SageMaker AI 承載範例。

如果您選擇 Amazon Bedrock，您會收到四個選項：

- 快速入門模型 - 快速入門一系列具有不同價格/效能特性的模型。建議用於建置您的第一個應用程式。此選項可讓您從提供的清單中選擇模型名稱。

- 其他基礎模型 - 使用不同的功能和專業能力存取完整範圍的基礎模型。此選項可讓您輸入所需 Bedrock 隨需基礎模型的模型 ID。
- 推論描述檔 - 推論描述檔利用 Bedrock 的跨區域推論，在尖峰使用率暴增期間將請求路由到多個 AWS 區域，以提高輸送量並改善彈性。此選項可讓您輸入要使用的推論設定檔 ID。
- 佈建模型 - 用於需要一致效能之生產工作負載的專用輸送量容量。此選項可讓您輸入要從 Amazon Bedrock 使用的佈建/自訂模型的 ARN。

模型選擇步驟也可讓您選擇進階模型設定。如需設定 Amazon Bedrock Guardrails、Amazon Bedrock 佈建輸送量和其他模型參數的詳細資訊，請參閱進階 LLM 設定。

跨區域推論

跨區域推論透過跨不同 AWS 區域的運算，協助 Amazon Bedrock 使用者無縫管理意外流量暴增。若要使用跨區域推論，您需要推論設定檔。推論描述檔是來自一組已設定 AWS 區域的隨需資源集區的抽象。它可以將源自您來源區域的推論請求路由到該集區中設定的另一個區域。這允許跨多個 AWS 區域的流量分佈。這有助於在尖峰需求期間實現更高的輸送量和增強的彈性。

推論描述檔是以其支援的模型和區域命名。您必須從其中包含的其中一個區域呼叫推論設定檔。例如，如下表所示，推論設定檔 ID `us.anthropic.claude-3-haiku-20240307-v1:0` 允許流量分佈到您選擇的模型的 `us-east-1`和 `us-west-2`區域。某些模型僅適用於特定區域中的推論設定檔。

推論設定檔	推論設定檔 ID	包含的區域
US Anthropic Claude 3 Haiku	`us.anthropic.claude-3-haiku-20240307-v1:0`	美國東部 (維吉尼亞北部) (`us-east-1`) 美國西部 (奧勒岡) (`us-west-2`)

如果您想要使用推論設定檔 ID 而非模型 ID，則必須識別適當的推論設定檔 ID。如需詳細資訊，請參閱《Amazon Bedrock 使用者指南》中推論設定檔的支援區域和模型。 在 Amazon Bedrock 主控台中，左側導覽選單中的跨區域推論選項會提供這些推論設定檔 IDs。

識別要使用的推論設定檔 ID 之後，您可以在選取模型階段執行下列步驟，以使用它：

1. 選取 Amazon Bedrock 做為模型提供者。
2. 選取推論設定檔選項按鈕選項。

3. 在出現的文字方塊中輸入您的推論設定檔 ID。

如需推論描述檔的詳細資訊，請參閱《Amazon Bedrock 使用者指南》中的使用跨區域推論改善彈性。

選取知識庫

如果您想要部署非擷取增強產生 (RAG) 使用案例，可以略過此步驟。

不過，如果您想要在部署中啟用 RAG，您現在可以提供預先設定的 Amazon Kendra 索引 ID 或 Amazon Bedrock 知識庫 ID。您也可以建立新的 Amazon Kendra 索引，以與 解決方案搭配使用。解決方案目前支援 Amazon Kendra 和 Amazon Bedrock 知識庫做為以 RAG 為基礎的使用案例部署的知識庫。

如需將資料擷取至知識庫以搭配 RAG 型部署使用的指導方針，請參閱設定知識庫一節。

進階 RAG 組態

精靈可讓您選取進階選項，以搭配 RAG 部署使用，例如每次將查詢傳送至知識庫時要擷取的文件數量、當知識庫中找不到文件時，LLM 的靜態文字回應、是否要顯示 LLM 回應的文件來源以進行安全檢查等。您也可以在搭配 Amazon Bedrock 知識庫使用 Amazon OpenSearch Serverless 時，為 Amazon Kendra 設定知識庫特定組態，例如角色型存取控制 (RBAC) 或覆寫搜尋類型。如需這些進階設定的詳細資訊，請參閱進階知識庫設定一節。

> **Note**
>
> 您的知識庫必須與部署的部署儀表板和使用案例堆疊位於相同的帳戶和區域。

選取提示和字符限制

在此步驟中，您可以設定提示以搭配 LLM 使用。提示可能需要預留位置，例如 `{input}`、`{history}`和 `{context}`。這些預留位置會指示 LLM 從何處繪製使用者輸入、對話歷史記錄，以及從知識庫擷取的資訊。

- 對於 Bedrock 模型提供者，必須提供對非 RAG 使用案例沒有限制的系統提示。但是 Bedrock 模型提供者的歧義提示需要至少兩個預留位置 - `{input}`和 `{history}`
- 對於 SageMaker 模型提供者、系統和歧義提示，兩個都需要至少兩個預留位置 - `{input}`和 `{history}`。
- 對於 RAG 使用案例，對於每個模型提供者，額外需要`{context}`預留位置。

如需詳細資訊，請參閱設定您的提示。您也可以參考管理模型字符限制的提示區段，同時為您的提示選取字符限制大小。

啟用多模式輸入

此步驟可讓您為使用案例啟用多模式輸入功能。啟用時，使用者可以上傳和傳送映像和文件及其文字查詢。

支援的檔案類型和限制條件：

- 影像：每則訊息最多 20 個影像。每個影像的大小不得超過 3.75 MB，高度和寬度不得超過 8，000 px。支援的格式：png、jpeg、gif、Webp
- 文件：每則訊息最多 5 個文件。每份文件的大小不得超過 4.5 MB。支援的格式：pdf、csv、doc、docx、xls、xlsx、html、txt、md

如何使用多模態輸入：

1. 在使用案例部署期間啟用 MultimodalEnabled 參數
2. 在聊天界面中，使用者可以以兩種方式上傳檔案：
   - 按一下聊天輸入方塊中的上傳按鈕，或
   - 直接將檔案拖放到聊天介面
3. 檔案會上傳至 Amazon S3，並由選取的模型處理
4. 上傳的檔案會在 48 小時後自動刪除

檔案狀態追蹤：

DevOps 使用者可以監控 DynamoDB 中的檔案中繼資料，其中包括上傳時間和處理狀態。檔案可以有下列狀態：

- 待定 - 檔案上傳已啟動但尚未完成。這是產生預先簽章 URL 時的初始狀態。
- 上傳 - 檔案已成功上傳至 S3，並準備好供模型處理。
- 已刪除 - 使用者已刪除檔案，且不應再存取以進行處理。
- 無效 - 檔案驗證檢查失敗（例如，檔案類型不符或安全驗證失敗）。

處於待定狀態且從未上傳的檔案會在 TTL 過期時自動清除。模型只能處理狀態為上傳的檔案。

S3 多模態儲存貯體和 DynamoDB 中繼資料表`MultimodalDataMetadataTable`分別在具有索引鍵 `MultimodalDataBucketName` 和 的部署儀表板輸出中提供。

> **Note**
>
> 並非所有模型都支援多模型輸入。在啟用此功能之前，請確定您選取的模型支援影像和文件處理。請參閱 Amazon Bedrock 文件中支援的基礎模型，以檢查哪個模型支援影像作為輸入模式。

> **Important**
>
> 使用者上傳的檔案會以 48 小時的生命週期政策存放在 Amazon S3 中。有關上傳檔案的中繼資料會存放在 Amazon DynamoDB 中，其中包含 24 小時的對話歷史記錄 TTL。

檢閱和部署

在此步驟之後，請檢閱您選取的設定，然後選擇部署使用案例。然後，新的使用案例會部署並在部署儀表板檢視中顯示，以進一步管理。

步驟 3b：部署 Bedrock 代理程式使用案例

Bedrock 代理程式使用案例提供強大且安全的機制，可在您的使用案例中叫用 Amazon Bedrock 代理程式。此功能可讓開發人員無縫整合採用 AI 技術的自主代理程式功能，在各種基礎模型、資料來源、軟體應用程式和使用者對話之間協調和執行多步驟任務，同時維持強大的安全措施。

先決條件

建立 Amazon Bedrock 代理程式之前，請確定您有下列項目：

1. 部署 AWS 上生成式 AI 應用程式建置器的 AWS 帳戶，可存取 Amazon Bedrock 主控台。
2. 建立和管理 Amazon Bedrock 代理程式的適當 IAM 許可。

建立 Amazon Bedrock 代理程式

如需建立代理程式的詳細說明，請參閱《Amazon Bedrock 使用者指南》中的手動建立和設定代理程式。 您可以設定選項，例如：

- 代理程式的指示（提示）

- 知識庫，用於根據使用者的輸入查詢其他資訊
- 客服人員的記憶體，可讓客服人員記住跨多個工作階段的資訊（最多 30 天）

成功建立 Amazon Bedrock 代理程式後，您可以繼續進行 AWS Bedrock 代理程式使用案例精靈流程上的生成式 AI 應用程式建置器。若要這麼做，請在部署儀表板上選擇部署新的使用案例，然後選取建立 Bedrock 代理程式使用案例。遵循精靈並使用下列步驟來設定使用案例。

選取使用案例

此步驟與上述文字使用案例相同。

選取網路組態

此步驟與上述文字使用案例相同

選取客服人員

在此步驟中，您必須提供您建立之 Amazon Bedrock 代理程式的代理程式 ID 和別名 ID。

步驟 3c：部署 MCP 伺服器使用案例

MCP（模型內容通訊協定）伺服器使用案例可讓您部署和管理可與 AI 模型和代理器整合的 MCP 伺服器。MCP 伺服器提供標準化的方式來向 AI 應用程式公開工具、資源和功能。您可以從現有的 Lambda 函數和 APIs 建立 MCP 伺服器，或使用容器映像託管自訂 MCP 伺服器。

先決條件

部署 MCP Server 使用案例之前，請確定您有下列項目：

1. 部署 AWS 上生成式 AI 應用程式建置器的 AWS 帳戶。
2. 建立和管理 Amazon Bedrock AgentCore 資源的適當 IAM 許可。
3. 根據您選擇的建立方法：
   - 對於閘道方法 (Lambda/API/MCP 伺服器）：Lambda 函數、API 端點及其對應的結構描述檔案 (Lambda 的 JSON 格式、APIs的 OpenAPI/Smithy) 或 MCP 伺服器 URL 端點
   - 對於執行期方法 (ECR)：將 Docker 容器映像推送至包含 MCP 伺服器實作的 Amazon ECR

MCP 伺服器建立方法

解決方案支援兩種建立 MCP 伺服器的方法：

從 Lambda、API 或 MCP 伺服器建立（閘道方法）

此方法會建立 MCP 閘道，以包裝現有的 Lambda 函數、REST APIs 或外部 MCP 伺服器，讓它們可以做為 MCP 工具存取。閘道會處理 MCP 與您現有服務之間的通訊協定轉譯。

- Lambda 目標：提供函數 ARN 和描述函數輸入/輸出格式的 JSON 結構描述檔案，以整合現有的 Lambda 函數
- OpenAPI 目標：使用 OpenAPI 規格 (JSON 或 YAML 格式） 整合 REST APIs，並支援 OAuth 2.0 或 API 金鑰身分驗證
- Smithy 目標：整合使用 Smithy 模型檔案 (.smithy 或 .json 格式） 定義的 APIs
- MCP 伺服器目標：透過 URL 端點直接連線至外部 MCP 伺服器，允許整合現有的 MCP 伺服器，而無需部署新的基礎設施

您可以在單一 MCP 閘道內設定多個目標（最多 10 個），每個目標都代表不同的工具或功能。

從 ECR Image 託管（執行期方法）

此方法會從 Amazon ECR 映像部署容器化 MCP 伺服器。當您的自訂 MCP 伺服器實作需要做為獨立服務執行時，請使用此方法。

- 提供 ECR 映像 URI（必須包含標籤，例如 `:latest`或 `:v1.0.0`)
- 選擇性地設定環境變數，以將組態傳遞至您的容器
- 容器必須實作 MCP 通訊協定並公開所需的端點

部署 MCP 伺服器

若要部署 MCP 伺服器使用案例，請在部署儀表板上選擇部署新的使用案例，然後選取建立 MCP 伺服器使用案例。遵循精靈並使用下列步驟來設定使用案例。

選取使用案例

此步驟與上述文字使用案例相同。

選取網路組態

目前僅啟用公有存取，且不支援 VPC 進行 netwrok 組態。

建立 MCP 伺服器

在此步驟中，您會設定 MCP 伺服器部署：

MCP 伺服器建立方法

選擇兩種建立方法：

- 從 Lambda、API 或 MCP 伺服器建立：從現有的 Lambda 函數、API 規格或外部 MCP 伺服器端點建立 MCP 閘道
- 從 ECR 映像託管：從容器映像部署自訂 MCP 伺服器

> **Note**
> 部署後無法變更建立方法。如果您需要切換方法，則必須部署新的 MCP 伺服器使用案例。

閘道組態（適用於 Lambda/API/MCP 伺服器方法）

如果您選取閘道方法，請設定一或多個目標：

1. 目標名稱（必要）：識別此目標組態的易記名稱
2. 目標描述（選用）：此目標功能的簡短描述
3. 目標類型：選取要設定的目標類型：
   - Lambda：適用於 AWS Lambda 函數
   - OpenAPI：適用於具有 OpenAPI 規格APIs REST API
   - Smithy：適用於具有 Smithy 模型定義的 APIs
   - MCP 伺服器：用於透過 URL 端點直接連線至外部 MCP 伺服器
4. 結構描述檔案（必要）：上傳描述目標的結構描述檔案：
   - 對於 Lambda：描述輸入/輸出格式的 JSON 結構描述檔案。如需建立 Lambda 工具結構描述的詳細資訊，請參閱《Amazon Bedrock AgentCore 開發人員指南》中的 Lambda 工具結構描述。
   - 對於 OpenAPI：OpenAPI 規格檔案 (JSON 或 YAML)。如需 OpenAPI 結構描述需求的詳細資訊，請參閱《Amazon Bedrock AgentCore 開發人員指南》中的 OpenAPI 結構描述。
   - 對於 Smithy：Smithy 模型檔案 (.smithy 或 .json)。如需建置 Smithy 目標的詳細資訊，請參閱《Amazon Bedrock AgentCore 開發人員指南》中的建置 Smithy 目標。
5. Lambda 函數 ARN (Lambda 目標需要）：要整合的 Lambda 函數 ARN
6. MCP 伺服器 URL (MCP 伺服器目標需要）：要連線之外部 MCP 伺服器的 URL 端點。URL 必須正確編碼，MCP 伺服器必須支援 MCP 通訊協定版本 2025-06-18 的工具功能。如需詳細資訊，請參閱《Amazon Bedrock AgentCore 開發人員指南》中的 MCP 伺服器目標。
7. 傳出身分驗證 (OpenAPI 目標需要）：設定 REST API 呼叫的身分驗證：

- 身分驗證類型：選擇 OAuth 2.0 或 API 金鑰
- 傳出身分驗證提供者 ARN：Amazon Bedrock AgentCore 字符文件庫中登入資料提供者的 ARN
- 其他組態：視身分驗證類型而定：
  - 對於 OAuth 2.0：設定範圍和自訂參數
  - 針對 API 金鑰：指定位置（標頭或查詢參數）、參數名稱和選用字首

您可以選擇新增另一個目標來新增多個目標（最多 10 個）。每個目標代表 MCP 伺服器公開的個別工具或功能。

ECR 組態（適用於 ECR Image 方法）

如果您選取執行期方法，請提供：

1. ECR 映像 URI（必要）：Amazon ECR 中 Docker 映像的完整 URI
   - 格式：`account-id.dkr.ecr.region.amazonaws.com/repository-name:tag`
   - 映像必須與部署位於相同的 AWS 區域
   - 需要標籤（例如 `:latest`、`:v1.0.0`)
2. 環境變數（選用）：設定金鑰/值對，以在執行時間傳遞至您的容器
   - 使用這些來提供組態、登入資料或自訂旗標
   - 您最多可以新增 10 個環境變數

檢閱和部署

設定 MCP 伺服器之後，請檢閱您選取的設定，然後選擇部署使用案例。然後，新的 MCP Server 使用案例會部署並在部署儀表板檢視中顯示，以便進一步管理。

> **Note**
>
> MCP Server 部署會在 Amazon Bedrock AgentCore 中建立資源，包括閘道、執行時間和工作負載身分。這些資源由 解決方案自動管理，並在您刪除使用案例時清除。

步驟 3d：部署 代理程式建置器使用案例

Agent Builder 可讓您在 Amazon Bedrock AgentCore 上建立、設定和部署生產就緒 AI 代理程式。此功能透過系統提示、模型選擇、MCP 伺服器整合和記憶體管理，提供對客服人員行為的完全控制。

部署程序主要與文字使用案例相同，但有一些顯著差異。

選取使用案例

此步驟與上述文字使用案例相同。

使用案例詳細資訊

此步驟與上述文字使用案例相同。

設定代理程式

在此步驟中，您會設定核心代理程式設定，包括系統提示、可用的 MCP 伺服器/字串工具和記憶體。

系統提示

系統提示會定義客服人員的行為、人格和功能。您可以：

- 編輯預設系統提示範本
- 使用重設為預設按鈕還原原始範本
- 包含工具用量和回應格式的說明

MCP 伺服器整合 （選用）

設定模型內容通訊協定伺服器，讓您的代理程式存取企業工具和資料：

1. 從下拉式清單中的可用 MCP 伺服器中選取
2. 檢閱可供客服人員存取的現成可用工具

> **Note**
>
> 在部署之前，必須設定和存取 MCP 伺服器。如需伺服器設定說明，請參閱 MCP 文件。

記憶體組態

設定代理程式如何維護內容和知識：

- 短期記憶體：預設為所有客服人員啟用。在工作階段中維護對話內容。
- 長期記憶體：切換以跨工作階段擷取和儲存洞見。使用 AgentCore 記憶體搭配語意記憶體策略。

檢閱和部署

在此步驟之後，請檢閱您選取的設定，然後選擇部署使用案例。客服人員建置器部署通常會在 10-15 分鐘內完成。然後，新的使用案例會在部署儀表板檢視中顯示，以便進一步管理。

步驟 3e：部署工作流程使用案例

工作流程建置器可讓您建立主管客服人員，使用客服人員做為工具委派模式來協調多個客服人員建置器客服人員。此功能可讓您重複使用現有的 Agent Builder 部署來建置複雜的多代理程式工作流程。

部署程序遵循與客服人員建置器類似的模式，以及客服人員探索和選擇的其他步驟。

選取使用案例

此步驟與上述文字使用案例相同。

使用案例詳細資訊

此步驟與上述文字使用案例相同。

設定主管代理程式

在此步驟中，您會設定主管客服人員，以協調專門的客服人員建置器客服人員。

系統提示

系統提示會定義主管客服人員委派給專業客服人員的運作方式。您可以：

- 編輯預設系統提示範本
- 包含客服人員選擇和委派的指示
- 定義如何彙總多個客服人員的結果
- 使用重設為預設按鈕還原原始範本

> **Note**
>
> 系統提示應清楚描述何時及如何使用每個專用代理程式。客服人員描述對於適當的委派至關重要。

模型選擇

選取主管客服人員的基礎模型。主管客服人員使用此模型來：

- 了解使用者請求
- 選取適當的專業客服人員
- 協調代理程式執行
- 彙總和格式化回應

選取專業客服人員

在此步驟中，您可以選取主管可以委派工作的客服人員建置器客服人員。

新增代理程式

1. 按一下新增客服人員以開啟客服人員選擇對話方塊
2. 從清單中選擇一或多個客服人員建置器客服人員
3. 檢閱將提供給主管的客服人員描述
4. 確認選擇

> **Note**
>
> - 工作流程需要至少 1 個客服人員建置器使用案例，做為專門的客服人員
> - 在建立工作流程之前，必須成功部署所有專門代理程式

檢閱和部署

檢閱工作流程組態，包括：

- 主管客服人員系統提示和模型
- 專門代理程式的清單
- 記憶體設定

選擇部署使用案例。工作流程部署通常會在 15-20 分鐘內完成。部署儀表板檢視中會顯示新的工作流程，以便進一步管理。

步驟 4：部署後組態

本節提供部署後設定解決方案的建議。

Amazon S3 儲存貯體版本控制、生命週期政策和跨區域複寫

此解決方案不會在其建立的儲存貯體上強制執行生命週期組態。我們建議下列作法：

- 設定生產部署的生命週期組態。如需詳細資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的在儲存貯體上設定生命週期組態。
- 根據部署解決方案的使用案例，啟用 Amazon S3 儲存貯體的版本控制和跨區域複寫。

Amazon DynamoDB 備份

此解決方案將 DynamoDB 用於多種用途（請參閱此解決方案中的 AWS 服務）。解決方案不會為其建立的資料表啟用備份。建議您為生產部署建立此功能的備份。如需詳細資訊，請參閱備份 DynamoDB 資料表和使用 DynamoDB 的 AWS Backup。

Amazon CloudWatch 儀表板和警示

解決方案會在 CloudWatch 中部署自訂儀表板，以轉譯自訂發佈指標和 AWS 服務指標的圖表。我們建議您建立 CloudWatch 警示，並根據部署解決方案的使用案例新增通知。

Amazon CloudWatch Logs

Lambda 日誌設定為永不過期，而 API Gateway 日誌設定為過期 10 年。您可以更新個別日誌群組的到期時間，以符合企業的記錄保留政策。

具有 TLS v1.2 或更新版本憑證的自訂 Web 網域

解決方案使用 CloudFront 部署 Web UI 和 Edge Optimized API Gateway。CloudFront 的網域不會強制執行 TLS v1.2 或更高版本的憑證。我們建議您使用 Amazon Route 53 建立自訂網域、使用 AWS Certificate Manager 建立憑證，或如果您的組織有現有憑證，請使用現有憑證。

如需其他詳細資訊，請參閱 Amazon Route 53 開發人員指南，並在 API Gateway 中選擇自訂網域的最低 TLS 版本。

使用 Amazon Kendra 擴展

此解決方案可讓您使用 Amazon Kendra 跨擷取的文件執行 NLP 支援的智慧型搜尋。對於較大的工作負載，您可以使用下列 CloudFormation 參數來增加 Amazon Kendra 的容量：

參數	預設	Description
Amazon Kendra 額外查詢容量	0	索引的額外查詢容量和 GetQuerySuggestions 容量。索引的額外容量單位每天提供大約 8，000 個查詢。
Amazon Kendra 額外儲存容量	0	索引的額外儲存容量數量。單一容量單位提供 30 GB 的儲存空間或 100，000 個文件，以先達到者為準。
Amazon Kendra 版本	Developer	Amazon Kendra 提供開發人員和企業版本來建立索引。如需 Amazon Kendra 版本之間差異的詳細資訊，請參閱 Amazon Kendra 定價。

若要修改這些 CloudFormation 參數的值，請在堆疊部署時選取適當的值。如需查詢和儲存容量單位的詳細資訊，請參閱調整容量。

> **Note**
>
> 如果未在啟用 RAG 的情況下部署文字使用案例，則不會使用或建立 Amazon Kendra 索引。

使用 Idp 聯合設定 SSO

此解決方案允許與支援 SAML 或 OIDC 型聯合身分的外部身分提供者整合。當解決方案部署時，它會為部署儀表板和個別使用案例建立 Amazon Cognito 使用者集區和個別應用程式用戶端整合。根據外部 Idp，請遵循 Amazon Cognito 開發人員指南中為使用者集區設定身分提供者一節中提供的步驟，然後選擇部署儀表板的應用程式用戶端整合，或您想要使用的使用案例來設定 SSO。

若要將使用者群組資訊傳遞至 RAG 架構中的知識庫或向量存放區，您需要將使用者群組從外部 Idp 映射至 Amazon Cognito 使用者群組。解決方案提供初始的堆疊 Lambda 函數觸發條件，以對應權杖產生前階段。Lambda 函數具有 group_mapping.json 檔案，必須更新以提供群組映射。請參閱使用

Amazon Cognito 支援的 Lambda 觸發條件的 Lambda 觸發條件自訂使用者集區工作流程。 Amazon Cognito

手動使用者集區組態

如果您選擇在部署期間不傳遞管理員或預設使用者電子郵件，則必須在 Amazon Cognito 中手動建立適當的使用者群組，以確保正確的許可：

1. 在部署儀表板中，在您的 Cognito 使用者集區`Admin`中建立名為 的群組。
2. 對於每個使用案例，在您的 Cognito 使用者集`${UseCaseName}-Users`區中建立名為 的群組，其中 `${UseCaseName}`是您部署的使用案例的名稱。

這些群組是授權機制正常運作的必要項目。您想要授予存取權的任何使用者都必須新增至適當的群組。

如果傳遞 `placeholder@example.com` ，將會建立 Cognito 群組，但您仍然必須建立相關聯的使用者，並將其指派給群組。

自訂登入畫面

此解決方案使用 Amazon Cognito 託管 UI 轉譯登入頁面。若要自訂內建登入頁面，請參閱《Amazon Cognito 開發人員指南》中的自訂內建登入和註冊網頁。

其他安全考慮事項

根據您部署解決方案的使用案例，檢閱下列安全建議：

- 客戶受管 AWS KMS 加密金鑰 - 解決方案預設使用 AWS 受管 AWS KMS 金鑰，因為這些金鑰無需額外費用。檢閱您的使用案例，判斷您是否應該更新解決方案以使用客戶受管的 AWS KMS 金鑰。
- API Gateway 限流規則 - 解決方案會使用 API Gateway 上的預設限流規則進行部署。根據您的使用案例和預期的交易量，我們建議您為 APIs 設定限流。如需詳細資訊，請參閱《Amazon API Gateway API Gateway 開發人員指南》中的調節 API 請求以提高輸送量。
- 啟用 AWS CloudTrail - 作為建議的安全實務，請考慮在部署解決方案的 AWS 帳戶中啟用 AWS CloudTrail，以在 AWS 帳戶中記錄 API 呼叫。如需詳細資訊，請參閱 AWS CloudTrail 使用者指南。
- 偏離偵測 - 我們建議在 CloudFormation 堆疊上設定偏離偵測，以識別已部署解決方案堆疊的意外或惡意變更並收到通知。如需詳細資訊，請參閱實作警示以自動偵測 AWS CloudFormation 堆疊中的偏離。

- Cognito JSON Web Token JWTs) - 解決方案使用 Amazon Cognito 發行JWTs 來驗證 REST API 端點。我們針對 ID 字符和存取字符設定了五分鐘到期的解決方案。當使用者登出時，其產生新權杖的能力會遭到撤銷 (重新整理權杖會遭到撤銷）。不過，在目前權杖過期之前，對 API 端點的任何請求都將成功驗證，因為它們具有有效的權杖。檢閱使用案例的安全考量，並調整字符有效期間。

自訂生命週期政策：

對於生產部署，請根據您的保留需求檢閱和調整生命週期政策。請參閱《Amazon Simple Storage Service 使用者指南》中的在儲存貯體上設定生命週期組態。

多模式檔案儲存和生命週期

如果您為使用案例啟用多模式輸入功能 (MultimodalEnabled 設定為 Yes)，解決方案會建立 Amazon S3 儲存貯體來存放上傳的檔案，並建立 DynamoDB 資料表來追蹤檔案中繼資料。

預設生命週期政策：

- S3 檔案：48 小時後自動刪除
- DynamoDB 中繼資料：記錄會在 24 小時後過期 （對話歷史記錄 TTL)

安全考量：

- 檔案會依使用案例 ID、使用者 ID、對話 ID 和訊息 ID 進行分割，而檔案會以 UUID 名稱儲存。UUID 與檔案名稱的映射可在 DynamoDB 中繼資料資料表中找到
- 使用者只能存取自己對話中上傳的檔案
- 使用魔術數字偵測執行檔案類型驗證
- 我們建議啟用 Amazon GuardDuty Malware Protection for S3 掃描上傳的檔案是否有惡意內容

部署獨立文字使用案例

遵循本節中的step-by-step說明，設定解決方案並將其部署到您的帳戶。

部署時間：約 10-30 分鐘

1. 登入 AWS 管理主控台，然後選取按鈕以啟動您要部署的 CloudFront 範本。

BedrockChat.template	Launch solution
SageMakerChat.template	Launch solution

2. 根據預設，範本會在美國東部（維吉尼亞北部）區域啟動。若要在不同 AWS 區域中啟動解決方案，請使用主控台導覽列中的區域選擇器。

   注意：此解決方案使用 Amazon Kendra 和 Amazon Bedrock，目前尚未在所有 AWS 區域提供。如果使用這些功能，您必須在提供這些服務的 AWS 區域中啟動此解決方案。如需各區域的最新可用性，請參閱 AWS 區域服務清單。

3. 在建立堆疊 *頁面上，確認正確的範本 URL 位於 *Amazon S3 URL *文字方塊中，然後選擇 *下一步。

4. 在 *指定堆疊詳細資訊 *頁面上，為您的解決方案堆疊指派名稱。如需有關命名字元限制的資訊，請參閱《AWS Identity and Access Management 使用者指南》中的 IAM 和 STS 限制。

5. 在參數下，檢閱此解決方案範本的參數，並視需要修改這些參數。此解決方案使用下列預設值。

UseCaseUUID	*<_## input_>*	36 個字元長的 UUIDv4，用於識別應用程式內的此部署使用案例。
UseCaseConfigRecordKey	*<_## input_>*	記錄的對應索引鍵，其中包含聊天提供者 Lambda 在執行時間所需的組態。資料表中的記錄必須具有與此值相符的金鑰屬性，以及包含所需組態的組態屬性。如果使用中，此記錄將由部署平台填入。對於此使用案例的獨立部署，需要在 UseCaseConfigTableName 中定義的資料表中手動建立的項目。

UseCaseConfigTableName	*<_## input_>*	堆疊將在金鑰 UseCaseConfigRecordKey 讀取具有此名稱的資料表中的組態
ExistingRestApiId	（選用輸入）	要使用的現有 API Gateway REST API ID。如果未提供，則會建立新的 API Gateway REST API。從部署儀表板部署時通常會提供 。 注意：當您需要部署多個獨立使用案例時，使用現有 APIs 有助於減少資源重複並簡化 APIs 的管理。為獨立使用案例提供現有 APIs 時，您需負責確保 API 已設定具有預期模型的必要路由（含）。必要的預先設定/詳細資訊路由（在聊天期間擷取使用案例詳細資訊），以及選擇性設定 /feedback 路由（如果 FeedbackEnabled 設定為 Yes 以啟用 LLM 聊天回應的意見回饋集合）。此外，也必須提供 ExistingApiRootResourceId、ExistingCognitoUserPoolId 和 ExistingCognitoGroupPolicyTableName。

ExistingApiRootResourceId	（選用輸入）	要使用的現有 API Gateway REST API 根資源 ID。您可以在 API 的「資源」區段中選取根資源 (/)，從 AWS 主控台取得 REST API 根資源 ID。然後，資源 ID 會顯示在資源詳細資訊面板中。您也可以在 REST API 上執行描述 API 呼叫，以尋找根資源 ID。
FeedbackEnabled	No	如果設定為否，則部署的使用案例堆疊將無法存取意見回饋功能。
ExistingModelInfoTableName	（選用輸入）	資料表的 DynamoDB 資料表名稱，其中包含模型資訊和預設值。部署平台使用的 。如果省略，則會建立新的資料表來存放模型預設值。
DefaultUserEmail	`placeholder@example.com`	此使用案例的預設使用者電子郵件。建立此電子郵件的 Amazon Cognito 使用者以存取使用案例。如果未提供，則不會建立 Cognito 群組和使用者。您也可以使用 `placeholder@example.com` 來建立 群組，但不能使用 使用者。如需設定使用者集區的相關資訊，請參閱手動使用者集區組態。

ExistingCognitoUserPoolId	（選用輸入）	要驗證此使用案例的現有 Amazon Cognito 使用者集區的 UserPoolId。通常在從部署儀表板部署時提供，但在獨立部署此使用案例堆疊時可以省略。
CognitoDomainPrefix	（選用輸入）	如果您想要為 Cognito 使用者集區用戶端提供網域，請輸入值。如果您未提供值，部署將產生一個值。
ExistingCognitoUserPoolClient	（選用輸入）	提供使用者集區用戶端（應用程式用戶端）以使用現有的集區用戶端。如果您未提供使用者集區用戶端，則會建立新的使用者集區用戶端。只有在提供現有的使用者集區 ID 時，才能提供此參數。
ExistingCognitoGroupPolicyTableName	（選用輸入）	包含使用者群組政策的 DynamoDB 資料表名稱。這是由自訂授權方在使用案例的 API 上使用。一般而言，您可以在從部署平台部署時提供輸入，但在獨立部署此使用案例堆疊時可以省略輸入。
RAGEnabled	`true`	如果設為 true，則部署的使用案例堆疊會使用建立的 Amazon Kendra 索引來提供 RAG 功能。如果設定為 `false`，使用者會直接與 LLM 互動。

KnowledgeBaseType	Bedrock	用於 RAG 的知識庫類型。只有在 RAGEnabled 為 時才設定`true`。可以是 Bedrock 或 Kendra。 注意：只有在 RAGEnabled 為 true 時才相關。
ExistingKendraIndexId	（選用輸入）	用於使用案例的現有 Kendra 索引的索引 ID。如果未提供任何 ，且 KnowledgeBaseType 為 Kendra，則會為您建立新的索引。 注意：只有在 RAGEnabled 為 `true`且 KnowledgeBaseType 為 時才相關`Kendra`。
NewKendraIndexName	（選用輸入）	要為此使用案例建立的新 Kendra 索引名稱。只有在未提供 ExistingKendraIndexId 時才適用。 注意：只有在 RAGEnabled 為 true 且 KnowledgeBaseType 為 Kendra 時才相關。

NewKendraQueryCapacityUnits	0	要為此使用案例建立新 Amazon Kendra 索引的其他查詢容量單位。只有在未提供 ExistingKendraIndexId 時才適用，請參閱 CapacityUnitsConfiguration。 注意：只有在 RAGEnabled 為 `true`且 KnowledgeBaseType 為 時才相關`Kendra`。
NewKendraStorageCapacityUnits	`0`	要為此使用案例建立新 Amazon Kendra 索引的額外儲存容量單位。只有在未提供 ExistingKendraIndexId 時才適用，請參閱 CapacityUnitsConfiguration。 注意：只有在 RAGEnabled 為 `true`且 KnowledgeBaseType 為 時才相關`Kendra`。
NewKendraIndexEdition	（選用輸入）	要針對此使用案例建立新 Amazon Kendra 索引的 Amazon Kendra 版本。只有在未提供 ExistingKendraIndexId 時才適用，請參閱 Amazon Kendra Editions。 注意：只有在 RAGEnabled 為 `true`且 KnowledgeBaseType 為 時才相關`Kendra`。

BedrockKnowledgeBaseId	（選用輸入）	要在 RAG 使用案例中使用的底端知識庫 ID。如果提供 ExistingKendraIndexId 或 NewKendraIndexName，則無法提供。 注意：只有在 RAGEnabled 為 `true`且 KnowledgeBaseType 為 時才相關`Bedrock`。
VpcEnabled	No	堆疊資源是否部署在 VPC 中。
CreateNewVpc	No	如果您希望解決方案為您建立新的 VPC，並用於此使用案例`Yes`，請選取 。 注意：只有在 VpcEnabled 為時才相關`Yes`。
IPAMPoolId	（選用輸入）	如果您想要使用 Amazon VPC IP Address Manager 指派 CIDR 範圍，請提供要使用的 IPAM 集區 ID。 注意：只有在 VpcEnabled 為 `Yes`且 CreateNewVpc 為 時才相關No。
ExistingVpcId	（選用輸入）	用於使用案例的現有 VPC VPC ID。 注意：只有在 VpcEnabled 為 `Yes`且 CreateNewVpc 為 時才相關No。

ExistingPrivateSubnetIds	（選用輸入）	逗號分隔的現有私有子網路 IDs 清單，用於部署 Lambda 函數。 注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時才相關No。
ExistingSecurityGroupIds	（選用輸入）	用於設定 Lambda 函數的現有 VPC 安全群組逗號分隔清單。 注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時才相關No。
VpcAzs	（選用輸入）	以逗號分隔的 AZs 清單，其中會建立 VPCs 的子網路 注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時才相關No。
UseInferenceProfile	No	如果設定的模型是 Bedrock，您可以指出是否使用 Bedrock 推論設定檔。這將確保必要的 IAM 政策將在堆疊部署期間設定。如需詳細資訊，請參閱下列 https：//https://docs.aws.amazon.com/bedrock/latest/userguide/cross-region-inference.html
DeployUI	是	選取選項以部署此部署的前端 UI。選取否， 只會建立託管 APIs的基礎設施、APIs的身分驗證，以及後端處理。

6. 選擇下一步。

7. 在 Configure stack options (設定堆疊選項) 頁面，選擇 Next (下一步)。
8. 在檢視 頁面上，檢視和確認的設定。選取確認範本將建立 AWS Identity and Access Management (IAM) 資源的方塊。
9. 選擇 Create stack (建立堆疊) 以部署堆疊。

您可以在狀態欄的 AWS CloudFormation 主控台中檢視堆疊的狀態。您應該會在大約 10-30 分鐘內收到 CREATE_COMPLETE 狀態。

部署獨立的 Bedrock 代理程式使用案例

遵循本節中的step-by-step說明，設定解決方案並將其部署到您的帳戶。

部署時間：約 10-30 分鐘

1. 登入 AWS 管理主控台，然後選取按鈕以啟動 CloudFront 範本。

BedrockAgent.template	Launch solution

2. 根據預設，範本會在美國東部（維吉尼亞北部）區域啟動。若要在不同 AWS 區域中啟動解決方案，請使用主控台導覽列中的區域選擇器。

> **Note**
>
> 此解決方案使用 Amazon Bedrock，目前尚未在所有 AWS 區域提供。如果您使用這些功能，您必須在提供這些服務的 AWS 區域中啟動此解決方案。如需各區域的最新可用性，請參閱 AWS 區域服務清單。

3. 在建立堆疊頁面上，確認正確的範本 URL 位於 Amazon S3 URL 文字方塊中，然後選擇下一步。
4. 在指定堆疊詳細資訊頁面上，為您的解決方案堆疊指派名稱。如需有關命名字元限制的資訊，請參閱 AWS Identity and Access Management 使用者指南中的 {https---docs-aws-amazon-com-https---docs-aws-amazon-com-IAM-latest-UserGuide-reference-iam-limits-html}【IAM 和 AWS STS 配額】。
5. 在參數下，檢閱此解決方案範本的參數，並視需要修改這些參數。此解決方案使用下列預設值。

參數	預設項目	Description
UseCaseUUID	*<_## input_>*	36 個字元長的 UUIDv4，用於識別應用程式內的此部署使用案例。
UseCaseConfigRecordKey	*####*	與記錄對應的索引鍵，其中包含聊天提供者 Lambda 函數在執行時間所需的組態。 資料表中的記錄必須具有與此值相符的金鑰屬性，以及包含所需組態的組態屬性。 如果此記錄正在使用，則將由部署平台填入。對於此使用案例的獨立部署，需要在 UseCaseConfigTableName 中定義的資料表中手動建立的項目。
UseCaseConfigTableName	*<## input>`*	堆疊將從此處提供的資料表讀取使用案例組態，並使用 UseCaseConfigRecordKey 中定義的記錄金鑰。
DefaultUserEmail	`placeholder@example.com`	此使用案例的預設使用者電子郵件。解決方案會為此電子郵件建立 Amazon Cognito 使用者，以存取使用案例。

參數	預設項目	Description
ExistingRestApiId	（選用輸入）	要使用的現有 API Gateway REST API ID。如果未提供，則會建立新的 API Gateway REST API。從部署儀表板部署時通常會提供 。 注意：當您需要部署多個獨立使用案例時，使用現有 APIs 有助於減少資源重複並簡化 APIs 的管理。為獨立使用案例提供現有 APIs 時，您需負責確保 API 已設定具有預期模型的必要路由（含）。必要的預先設定/詳細資訊路由（在聊天期間擷取使用案例詳細資訊），以及選擇性設定 /feedback 路由（如果 FeedbackEnabled 設定為 Yes 以啟用 LLM 聊天回應的意見回饋集合）。此外，也必須提供 ExistingApiRootResourceId、ExistingCognitoUserPoolId 和 ExistingCognitoGroupPolicyTableName。
ExistingApiRootResourceId	（選用輸入）	要使用的現有 API Gateway REST API 根資源 ID。在 API 的「資源」區段中選取根資源 (/)，即可從 AWS 主控台取得 REST API 根資源 ID。資源 ID 隨即會顯示在資源詳細資訊面板中。您也可以在 REST API 上執行描述 API 呼叫，以尋找根資源 ID。

參數	預設項目	Description
FeedbackEnabled	No	如果設定為否，則部署的使用案例堆疊將無法存取意見回饋功能。
CognitoDomainPrefix	（選用輸入）	如果您想要為 Amazon Cognito 使用者集區用戶端提供網域，請輸入值。如果您不提供值，解決方案會產生一個值。
ExistingCognitoUserPoolId	（選用輸入）	您要驗證此使用案例的現有 Amazon Cognito 使用者集區的 UserPoolId。注意：您通常在從部署儀表板部署時提供此 ID，但在獨立部署此使用案例堆疊時可以省略它。
ExistingCognitoUserPoolClient	（選用輸入）	提供使用者集區用戶端（應用程式用戶端）以使用現有的集區用戶端。如果您不提供使用者集區用戶端，解決方案會建立一個。只有在您提供 ExistingCognitoUserPoolId 時，才能提供此參數。
ExistingCognitoGroupPolicyTableName	（選用輸入）	包含使用者群組政策的 DynamoDB 資料表名稱。這是由自訂授權方在使用案例的 API 上使用。注意：您通常在從部署儀表板部署時提供此名稱，但在獨立部署此使用案例堆疊時可以省略它。
VpcEnabled	No	堆疊資源是否部署在 VPC 中。

參數	預設項目	Description
CreateNewVpc	No	Yes 如果您希望解決方案為您建立新的 VPC，並將其用於此使用案例，請選取 。注意：只有在 VpcEnabled 為 時，此參數才相關Yes。
IPAMPoolId	（選用輸入）	如果您想要使用 IPAM 指派 CIDR 範圍，請提供要使用的 IPAM 集區 ID。注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時，此參數才相關No。
ExistingVpcId	（選用輸入）	要用於使用案例之現有 VPC 的 VPC ID。注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時，此參數才相關No。
ExistingPrivateSubnetIds	（選用輸入）	逗號分隔的現有私有子網路 IDs 清單，用於部署 Lambda 函數。注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時，此參數才相關No。
ExistingSecurityGroupIds	（選用輸入）	以逗號分隔的現有 VPC 安全群組清單，用於設定 Lambda 函數。注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時，此參數才相關No。

參數	預設項目	Description
VpcAzs	（選用輸入）	以逗號分隔的 AZs 清單，其中會建立 VPCs 的子網路 注意：只有在 VpcEnabled 為 Yes且 CreateNewVpc 為 時才相關No。
BedrockAgentId	####	要使用的 Amazon Bedrock 代理程式 ID。
BedrockAgentAliasId	####	要使用的 Amazon Bedrock 代理程式別名 ID。
DeployUI	Yes	選取選項以部署此部署的前端聊天 UI。選取 No 會導致建立託管 APIs基礎設施、APIs的身分驗證，以及沒有聊天 UI 的後端處理。

6. 選擇下一步。
7. 在 Configure stack options (設定堆疊選項) 頁面，選擇 Next (下一步)。
8. 在檢視 頁面上，檢視和確認的設定。選取確認範本將建立 IAM 資源的方塊。
9. 選擇 Create stack (建立堆疊) 以部署堆疊。

您可以在狀態欄的 AWS CloudFormation 主控台中檢視堆疊的狀態。您應該會在大約 10-30 分鐘內收到 CREATE_COMPLETE 狀態。

提供 DynamoDB 聊天組態

部署使用案例時，UseCaseConfigRecordKey 和 UseCaseConfigTableName 是通常由部署儀表板填入的必要 CloudFormation 參數。部署儀表板堆疊會處理此資料表的建立和組態，而呼叫部署 API 觸發參數的人口。

執行獨立部署時，您必須執行下列動作：

1. 建立具有金鑰雜湊索引鍵的 DynamoDB 資料表。

2. 在包含使用案例組態的資料表中建立記錄，做為格式的記錄：`{key: some_use_case_key, config: {your_configuration}.`
3. 在部署時，將選擇的 UseCaseConfigTableName 和 UseCaseConfigRecordKey (`some_use_case_key`在此範例中為) 參數傳遞至使用案例堆疊。

若要為獨立部署建立適當的組態，您可以從部署儀表板建立必要的使用案例，並從組態資料表複製記錄。否則，您可以根據下列 Bedrock 部署範例來製作自己的組態：

```
{
 "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
```

```
  }
}
```

使用 Service Catalog AppRegistry 監控解決方案

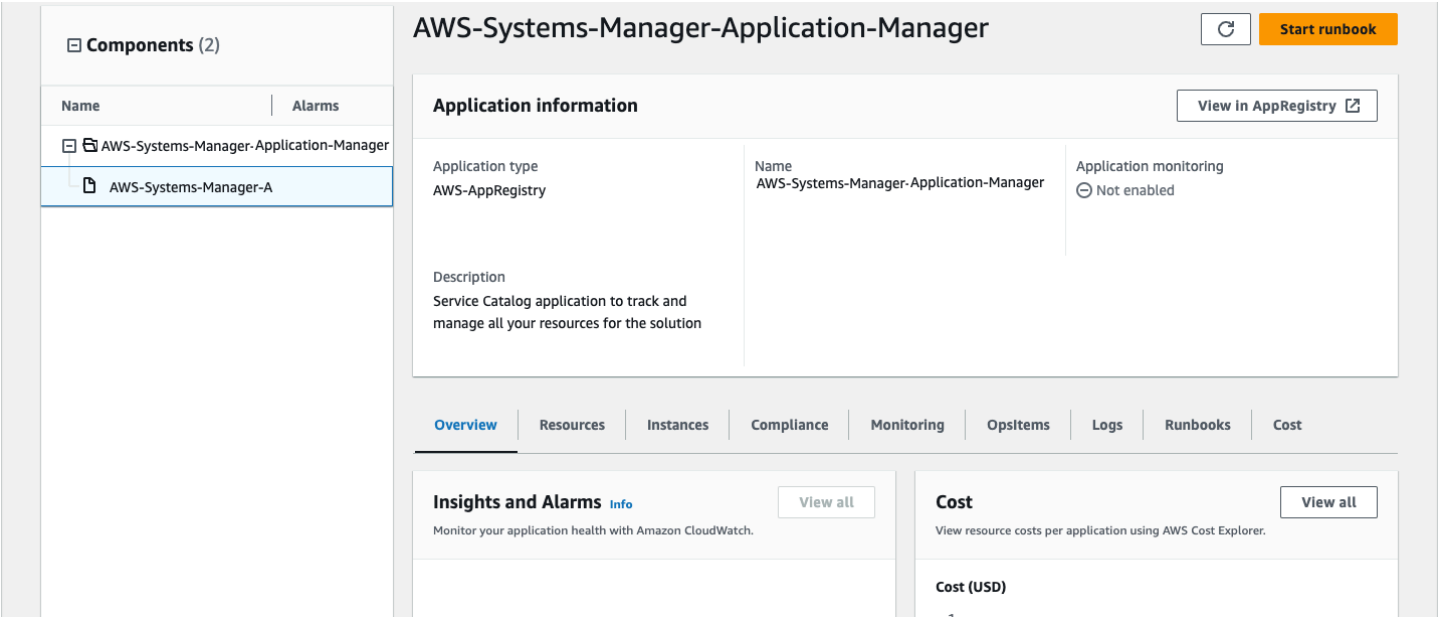
解決方案包含 Service Catalog AppRegistry 資源，可將 CloudFormation 範本和基礎資源註冊為 Service Catalog AppRegistry 和 Systems Manager Application Manager 中的應用程式。

Systems Manager Application Manager 為您提供此解決方案及其資源的應用程式層級檢視，以便您可以：

- 從中央位置監控其資源、跨堆疊和 AWS 帳戶部署資源的成本，以及與此解決方案相關聯的日誌。
- 在應用程式的內容中檢視此解決方案資源的操作資料。例如，部署狀態、CloudWatch 警示、資源組態和操作問題。

下圖說明 Application Manager 中解決方案堆疊的應用程式檢視範例。

Application Manager 中的描述解決方案堆疊



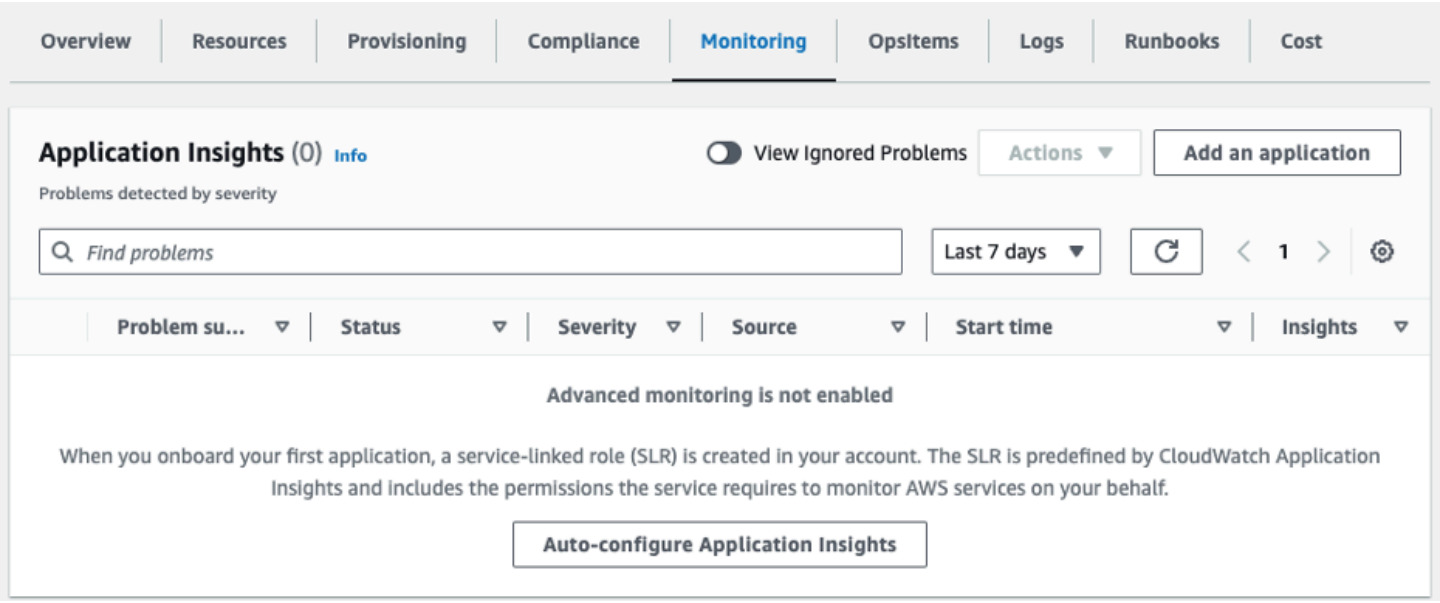
啟用 CloudWatch Application Insights

1. 登入 Systems Manager 主控台。
2. 在導覽窗格中，選擇 Application Manager。
3. 在應用程式中，搜尋此解決方案的應用程式名稱，然後選取它。

應用程式名稱在應用程式來源欄中會有應用程式登錄檔，而且會有解決方案名稱、區域、帳戶 ID 或堆疊名稱的組合。

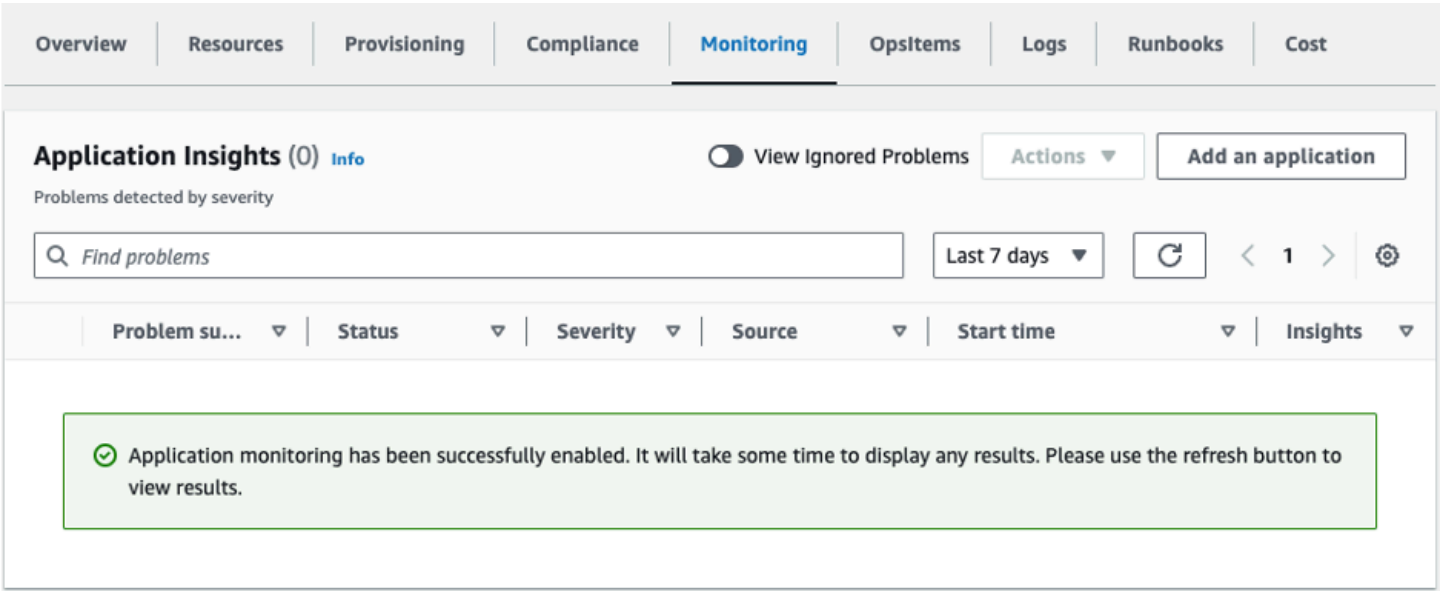
4. 在元件樹狀目錄中，選擇您要啟用的應用程式堆疊。
5. 在監控索引標籤的 Application Insights 中，選取自動設定 Application Insights。

Application Insights 儀表板未顯示偵測到的問題和自動設定的選項。



現在已啟用應用程式的監控，並顯示下列狀態方塊：

Application Insights 儀表板顯示成功的監控啟用訊息。



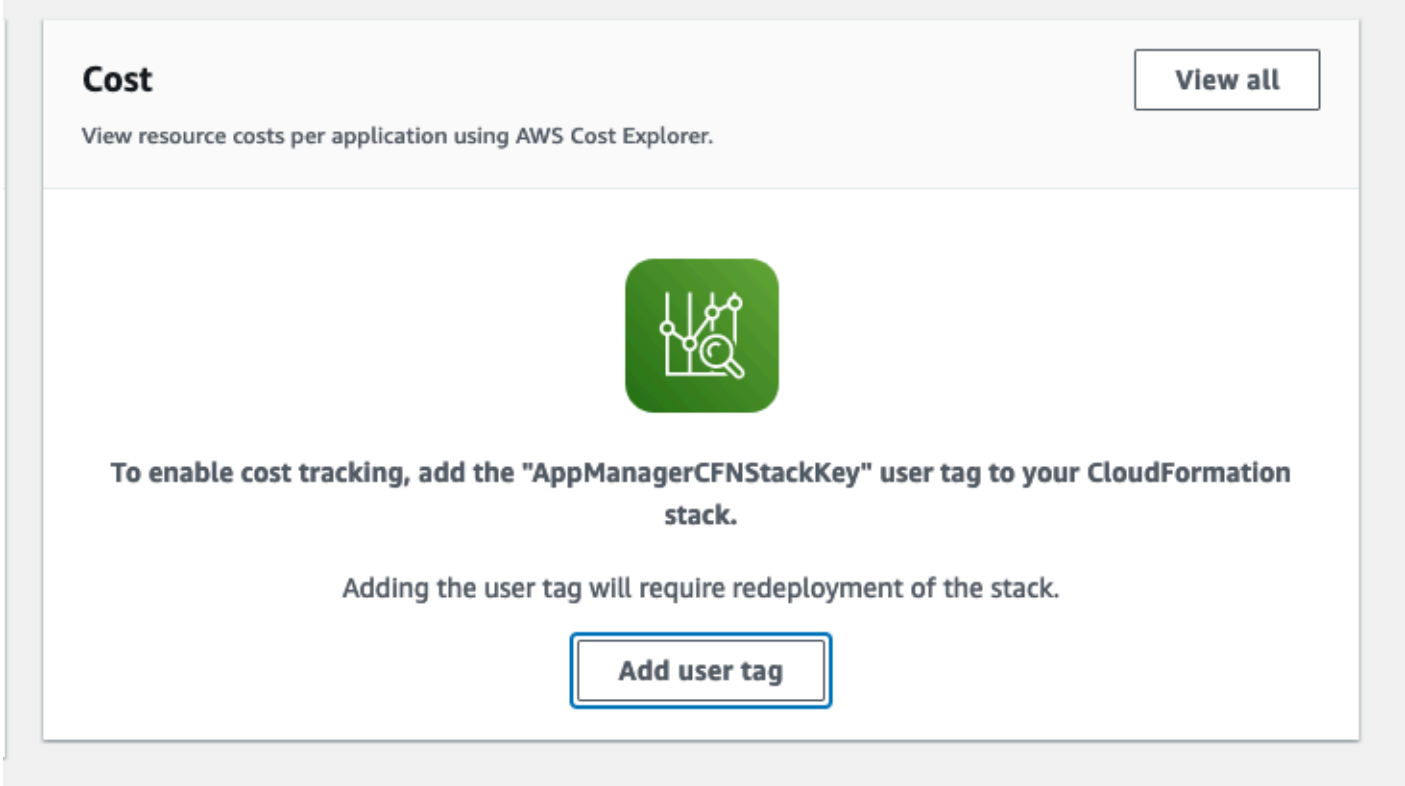
確認與解決方案相關聯的成本標籤

啟用與解決方案相關聯的成本分配標籤後，您必須確認成本分配標籤，以查看此解決方案的成本。若要確認成本分配標籤：

1. 登入 Systems Manager 主控台。
2. 在導覽窗格中，選擇 Application Manager。
3. 在應用程式中，選擇此解決方案的應用程式名稱，然後選取它。

   應用程式名稱在應用程式來源欄中會有應用程式登錄檔，而且會有解決方案名稱、區域、帳戶 ID 或堆疊名稱的組合。
4. 在概觀索引標籤的成本中，選取新增使用者標籤。

   描述 Application Cost 新增使用者標籤畫面的螢幕擷取畫面



5. 在新增使用者標籤頁面上，輸入 `confirm`，然後選取新增使用者標籤。

啟用程序最多可能需要 24 小時才能完成，並顯示標籤資料。

啟用與解決方案相關聯的成本分配標籤

啟用 Cost Explorer 之後，您必須啟用與此解決方案相關聯的成本分配標籤，才能查看此解決方案的成本。成本分配標籤只能從組織的管理帳戶啟用。若要啟用成本分配標籤：

1. 登入 AWS Billing and Cost Management and Cost Management 主控台。
2. 在導覽窗格中，選取成本分配標籤。
3. 在成本分配標籤頁面上，篩選 AppManagerCFNStackKey 標籤，然後從顯示的結果中選取標籤。
4. 選擇 Activate (啟用)。

AWS Cost Explorer

您可以透過與必須先啟用的 AWS Cost Explorer 整合，在 Application Manager 主控台中查看與應用程式和應用程式元件相關的成本概觀。Cost Explorer 透過提供一段時間內 AWS 資源成本和用量的檢視，協助您管理成本。若要為解決方案啟用 Cost Explorer：

1. 登入 AWS Cost Management 主控台。
2. 在導覽窗格中，選取 Cost Explorer 以檢視解決方案隨時間的成本和用量。

更新解決方案

如果您先前已部署解決方案，請依照此程序更新解決方案的 CloudFormation 堆疊，以取得最新的功能和增強功能。升級程序分為三個部分：

- 步驟 1：更新部署儀表板
- 步驟 2：遷移使用案例組態
- 步驟 3：更新使用案例

> **Note**
>
> 1. 在 2.0.0 版中，為了支援 Amazon Bedrock 和 Amazon SageMaker AI，已棄用與 Anthropic 和 Hugging Face 的整合。您可以透過 SageMaker JumpStart 部署 Hugging Face 提供的模型。如需詳細資訊，請參閱搭配使用 Hugging Face 與 Amazon SageMaker AI。
> 2. 在執行這些步驟之前，請務必在非生產環境中測試更新程序。

步驟 1：更新部署儀表板

1. 登入 CloudFormation 主控台，選取您現有的 CloudFormation 堆疊，然後選取更新。
2. 選取取代目前範本。
3. 在指定範本下：
   a. 選取 Amazon S3 URL。
   b. 複製最新的 CloudFormation 範本連結。
   c. 將連結貼到 Amazon S3 URL 方塊中。
   d. 驗證 Amazon S3 URL 文字方塊中顯示的範本 URL 是否正確，然後選擇下一步。再次選擇 Next (下一步)。
4. 在參數下，檢閱範本的參數並視需要修改。如需參數的詳細資訊，請參閱步驟 1：啟動部署儀表板堆疊。
5. 選擇下一步。
6. 在 Configure stack options (設定堆疊選項) 頁面，選擇 Next (下一步)。
7. 在檢視 頁面上，檢視和確認的設定。勾選確認範本將建立 IAM 資源的方塊。

8. 選擇檢視變更集並驗證變更。
9. 選擇更新堆疊以部署堆疊。

您可以在狀態欄的 AWS CloudFormation 主控台中檢視堆疊的狀態。您應該會在大約 10 分鐘內收到 UPDATE_COMPLETE 狀態。

如果現有的解決方案版本在 v2.0.0 之前，更新會建立 Web UI 堆疊（以 Cognito 託管 UI 取代登入畫面的`amplify-ui`實作）和新的 CloudFront URL，一旦堆疊狀態為 UPDATE_COMPLETE，即可從 CloudFormation 主控台的輸出區段取得。

> **Note**
>
> 在您完成下列步驟之前，不會顯示使用 v2.0.0 之前版本建立的現有使用案例。

步驟 2：遷移使用案例組態（僅更新低於 2.0.0 的版本）

2.0.0 版中儲存的結構描述和儲存使用案例組態的 AWS 服務已變更。使用 gaab_v2_migration.py 指令碼，遵循 GAAB v2 Migration 使用者指南中所述的步驟。 https://github.com/aws-solutions/generative-ai-application-builder-on-aws/blob/main/source/scripts/v2_migration/gaab_v2_migration.py執行指令碼後，您可以存取部署儀表板來檢視已部署的使用案例。

> **Note**
>
> 您必須依照下列步驟完成遷移使用案例。

步驟 3：更新使用案例

您可以使用最新版本的 GAAB 中提供的新功能來編輯已部署的使用案例。如需如何使用此解決方案中的功能的資訊，請參閱使用解決方案。

若要將使用案例更新至最新版本，您必須在部署儀表板中完成 `Edit`use case 步驟（雖然您可能不會進行任何變更）。此動作會觸發具有最新範本版本的 CloudFormation 堆疊更新。

> **Note**
>
> 使用 1.x 或 2.x 版解決方案建立的使用案例可能無法搭配更新版本使用。因此，我們建議您透過部署儀表板複製使用 v3.0.0 之前版本建立的現有使用案例。然後，逐步遷移並使用 v3.0.0 或更新版本建立的新使用案例取代 。

疑難排解

本節提供部署和使用 解決方案的疑難排解指示。

如果這些指示無法解決您的問題，請聯絡 Support 提供為此解決方案開啟支援案例的說明。

問題：使用為我建立 VPC 部署已啟用 VPC 的組態失敗

部署儀表板堆疊或使用案例堆疊失敗部署，因為 CloudFormation 無法佈建 VPC 網路資源。

Resolution

檢查您帳戶中 VPCs 和彈性 IPs配額限制。預設限制為每個 AWS 區域每個 AWS 帳戶彈性 IPs 和 VPCs 各 5 個。

> **Note**
>
> 當解決方案建立 VPC 時，單一啟用 VPC 的部署（部署儀表板或使用案例）是 2-AZ部署，每個可用區域都有 1 個公有和 1 個私有子網路，每個公有子網路都會部署 1 個 NAT 閘道。使用 2 個 NAT 閘道時，部署會使用配額限制中的 2 個公有 IP 地址。

需要注意的一些限制（每個帳戶、每個區域）：

- VPCs 數量 - 5
- 公有 IP 地址數量 - 5
- 閘道 VPC 端點的數量 - 20
- Inteface VPC 端點的數量 - 20

問題：在刪除部署儀表板堆疊之後，無法在 CloudFormation 中刪除使用案例堆疊

如果在刪除所有使用案例堆疊之前，在 CloudFormation 中刪除部署儀表板堆疊，則使用案例最終可能會處於鎖定（無法使用）狀態。這是因為部署儀表板堆疊建立的 IAM 角色不再存在，無法修改使用案例堆疊。

Resolution

> **Warning**
>
> 確保您在使用後立即清除任何手動建立的角色。這些是提升的許可，使用者可以利用這些許可來提升角色。

重新建立已刪除的 IAM 角色，以啟用 CloudFormation 堆疊的刪除：

1. 開啟 CloudFormation 主控台，並判斷與您鎖定堆疊相關聯的角色。
   a. 您可以在標記 IAM 角色的堆疊資訊區段中找到角色 ARN。
   b. 角色名稱是 IAM 角色 ARN 中 ：role/ 後面的內容 （ 例如 arn：aws：iam：：<account-id>：role/<role-name>)
2. 在 IAM 中建立與已刪除角色同名的新角色。
   a. 選取 AWS 服務做為信任的實體，然後從下拉式清單中選取 CloudFormation。
   b. 新增必要的許可。如果您不確定所需的許可，可以使用 AWS 受管 AdministratorAccess 政策。
   c. 輸入與步驟 1 中所取得完全相同的角色名稱。
3. 返回 CloudFormation 主控台並刪除鎖定的堆疊。
4. 成功刪除所有鎖定的堆疊後，請返回 IAM 並刪除步驟 2 中建立的任何角色。

問題：使用案例 UI 不會反映設定中的變更

更新使用案例時，UI 會部署到 CloudFront。不過，由於 CloudFront 會快取部署，以及指示使用者如何顯示某些設定的組態檔案，因此這些變更可能不會立即反映。

Resolution

CloudFront 分佈可以失效，以強制將新組態傳播給前端使用者。

1. 開啟 CloudFormation 主控台，並判斷與您的使用案例堆疊相關聯的 CloudFront 分佈。
   a. 使用案例堆疊應以您在部署使用案例時使用的相同名稱開頭。
   b. 找到對應至 UI 的巢狀堆疊。巢狀堆疊名稱應以 WebAppS3UINestedStackS3UINestedStackResource 開頭。

c. 在資源索引標籤下，找到 AWS::CloudFront::Distribution類型的資源，然後選取實體 ID。這將在 CloudFront 主控台中開啟分佈。

2. 導覽至失效索引標籤，然後選擇建立失效，然後輸入 /* 的路徑。這會使所有路徑失效。
3. 在您自己的瀏覽器中，刪除與使用案例相關的任何 Cookie 和快取檔案。

聯絡 AWS Support

如果您有 AWS Business Support+、AWS Enterprise Support 或 Unified Operations，您可以使用 AWS Support Center 取得此解決方案的專家協助。以下章節將提供說明。

建立案例

1. 登入支援中心。
2. 選擇建立案例。

如何提供協助？

1. 選擇技術。
2. 針對服務，選取解決方案。
3. 針對類別，選取其他解決方案。
4. 針對嚴重性，選取最符合您使用案例的選項。
5. 當您輸入服務、類別和嚴重性時，界面會填入常見故障診斷問題的連結。如果您無法使用這些連結來解決問題，請選擇下一步：其他資訊。

其他資訊

1. 針對主旨，輸入摘要您的問題的文字。
2. 針對描述，請詳細說明問題，包括此解決方案的名稱：AWS 上的生成式 AI 應用程式建置器。
3. 選擇連接檔案。
4. 連接 AWS Support 處理請求所需的資訊。

協助我們更快解決您的案例

1. 輸入請求的資訊。
2. 選擇下一步驟：立即解決或聯絡我們。

立即解決或聯絡我們

1. 檢閱立即解決解決方案。
2. 如果您無法解決這些解決方案的問題，請選擇聯絡我們，輸入請求的資訊，然後選擇提交。

解除安裝解決方案

> **Note**
>
> 透過 部署儀表板建立的部署不適用於在 解決方案之外進行管理。在 CloudFormation 中刪除堆疊之前，請務必從部署儀表板中刪除和清除任何部署。

您可以從 AWS 管理主控台或使用 AWS 命令列界面解除安裝 AWS 解決方案上的生成式 AI 應用程式建置器。您必須手動刪除此解決方案建立的 Amazon S3 儲存貯體、Amazon Kendra 索引或 CloudWatch Logs。如果您有要保留的儲存資料，AWS 解決方案不會自動刪除 Amazon S3 儲存貯體、Amazon Kendra 索引或 CloudWatch Logs。

使用 AWS 管理主控台

1. 登入 AWS CloudFormation 主控台。
2. 在堆疊頁面上，選取此解決方案的安裝堆疊。
3. 選擇 刪除。

使用 AWS 命令列界面

判斷您的環境中是否可使用 AWS Command Line Interface (AWS CLI)。如需安裝說明，請參閱《AWS CLI 使用者指南》中的什麼是 AWS 命令列界面。 確認 AWS CLI 可用後，請執行下列命令。

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

手動解除安裝步驟

刪除 Amazon S3 儲存貯體

如果您決定刪除 AWS CloudFormation 堆疊，此解決方案會設定為保留解決方案建立的 Amazon S3 儲存貯體，以防止意外資料遺失。解除安裝解決方案之後，如果您不需要保留資料，您可以手動刪除此 Amazon S3 儲存貯體。請依照下列步驟刪除 Amazon S3 儲存貯體。

1. 登入 Amazon S3 主控台。

2. 在導覽窗格中，選取儲存貯體。
3. 找到 <stack-name> S3 儲存貯體。
4. 選取 S3 儲存貯體，然後選擇刪除。

若要使用 AWS CLI 刪除 S3 儲存貯體，請執行下列命令。使用 --force 選項時，您不需要先清空儲存貯體。

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

刪除 Amazon Kendra 索引

為了防止意外資料遺失，此解決方案設定為在刪除 AWS CloudFormation 堆疊時保留解決方案建立的 Amazon Kendra 索引。解除安裝解決方案之後，您可以手動刪除不再需要保留資料的 Amazon Kendra 索引。請依照下列步驟刪除 Amazon Kendra 索引。

1. 登入 Amazon Kendra 主控台。
2. 在導覽窗格中，選取索引。
3. 找到並選取您要刪除的索引。
4. 選取刪除以刪除所選索引。

若要使用 AWS CLI 刪除 Amazon Kendra 索引，請執行下列命令：

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

刪除 CloudWatch Logs

為了防止意外資料遺失，如果您決定刪除 CloudFormation 堆疊，我們會將此解決方案設定為保留 CloudWatch Logs。 CloudFormation 解除安裝解決方案之後，如果您不需要保留資料，您可以手動刪除日誌。請依照下列步驟刪除 CloudWatch Logs。

1. 登入 Amazon CloudWatch 主控台。
2. 在導覽窗格中，選取日誌群組。
3. 找出解決方案建立的日誌群組。
4. 選取其中一個日誌群組。
5. 選擇 Actions (動作)，然後選擇 Delete (刪除 VPC)。

重複這些步驟，直到您刪除所有解決方案日誌群組為止。

使用 解決方案

存取 UI

在堆疊部署程序期間（適用於部署儀表板和使用案例），電子郵件會傳送至設定的電子郵件地址。電子郵件包含使用者可用來註冊和存取 Web 介面的臨時登入資料。

> **Note**
>
> 有權存取 AWS 管理主控台的 DevOps 使用者，必須在堆疊完成時為管理員使用者提供部署儀表板 UI 的 CloudFront URL。

對於使用案例，具有部署儀表板 UI 存取權的管理員使用者，必須在部署完成時為商業使用者提供使用案例 UI 的 CloudFront URL。

登入後，使用者可以與解決方案 UIs 互動，在管理員情況下為部署儀表板，或在商業使用者情況下為使用案例。

如何更新部署

在部署儀表板首頁（或部署的詳細資訊頁面）上，您可以編輯部署所使用的組態。您只能編輯處於 CREATE_COMPLETE 或 UPDATE_COMPLETE 狀態的部署。

除了使用案例名稱之外，所有其他選項都可以編輯部署。只要變更您要編輯並重新部署的值即可。

根據所做的編輯範圍，重新部署時間會有所不同。如果簡易設定已變更（例如，模型參數），則可能需要幾秒鐘的時間，如果較大的基礎設施相關選項已變更，則可能需要超過 30 分鐘的時間（例如，請求為文字使用案例 RAG 建立 Amazon Kendra 索引）。

編輯成功完成後，應用程式狀態將報告 UPDATE_COMPLETE 狀態。目前，您可以透過 CloudFront URL 存取部署的 UI，並與修改後的部署互動。

> **Note**
>
> 如果您想要比較不同的設定或 LLMs，可以更輕鬆地side-by-side執行多個部署。使用複製功能快速使用現有的組態來啟動新的部署。

如何複製部署

在部署儀表板首頁（或部署的詳細資訊頁面）上，您可以複製部署所使用的組態。複製部署會啟動部署新的使用案例精靈，但大多數欄位會預先填入相同的值。

這是一項方便的操作，可協助您快速複製具有已變更設定的部署、復原已刪除的部署，或比較其他相同部署中的多個 LLMs。

如何刪除部署

在部署儀表板首頁（或部署的詳細資訊頁面）上，您可以在不再需要部署時將其刪除。刪除部署會叫用 CloudFormation 堆疊刪除操作，並取消佈建部署的資源。

根據預設，已刪除的部署仍會保留在儀表板上，以啟用複製功能。若要從儀表板完全移除部署，以便在 UI 中停止追蹤，請在刪除確認視窗中選擇永久刪除。

> **Important**
>
> 某些資源會在堆疊刪除期間保留，且必須手動刪除。如需保留哪些資源以及如何清除這些資源的詳細資訊，請參閱手動解除安裝一節。

設定大型語言模型 (LLM)

哪種 LLM 適用於您的使用案例，取決於您需求所需的大量因素，以及您想要策劃的客戶體驗類型。此解決方案看起來並不具有規範性，而是旨在為您提供必要的工具，以評估什麼最適合您的應用程式。

AI 產生的空間正在快速發展，因此您必須隨時掌握最新的模型、最佳化技術和最佳實務，以確保您為客戶打造正確的體驗。

> **Note**
>
> 如果您使用的是非公有或敏感資料，請務必使用 AWS 服務（例如 Amazon Bedrock 或 Amazon SageMaker AI) 選取 LLM 選項。與使用由第三方供應商託管的 LLM 相比，這透過在您的 區域和 AWS 網路上保留資料來改善部署的整體安全狀態。

使用 Amazon SageMaker AI 做為 LLM 供應商

從 v1.3.0 開始，Amazon SageMaker AI 可作為文字使用案例的模型提供者。此功能可讓您使用解決方案中 AWS 帳戶內已存在的 SageMaker AI 推論端點。以下是一些開始使用的方法。

> **Important**
>
> 解決方案不會管理 SageMaker AI 端點的生命週期。您負責刪除不再需要的 SageMaker AI 端點，以停止產生額外費用。

建立 SageMaker AI 端點

您可以使用 Amazon SageMaker AI JumpStart 快速部署端點。

您也可以使用文字產生的 SageMaker AI 端點，並使用基本 SageMaker AI 服務進行部署。如需如何部署模型以進行推論的逐步指南，請參閱 SageMaker AI JumpStart 文件。

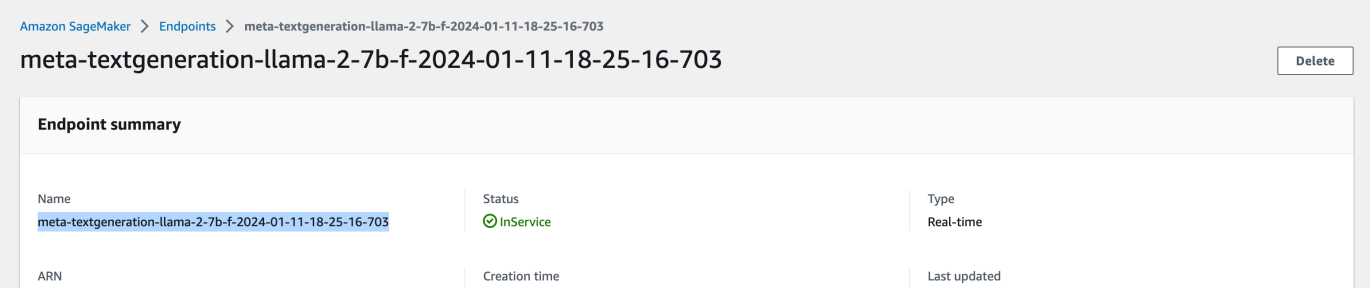
> **Note**
>
> 基礎模型/LLMs 通常相當大，通常需要使用大型加速運算執行個體。根據預設，您的 AWS 帳戶中可能無法使用這些較大的執行個體。請參閱預設 SageMaker AI 配額，並確保在部署之前請求增加配額，以避免常見的部署失敗。

使用 SageMaker AI 端點建立文字使用案例部署

若要使用 SageMaker AI 端點部署新的文字使用案例以進行推論：

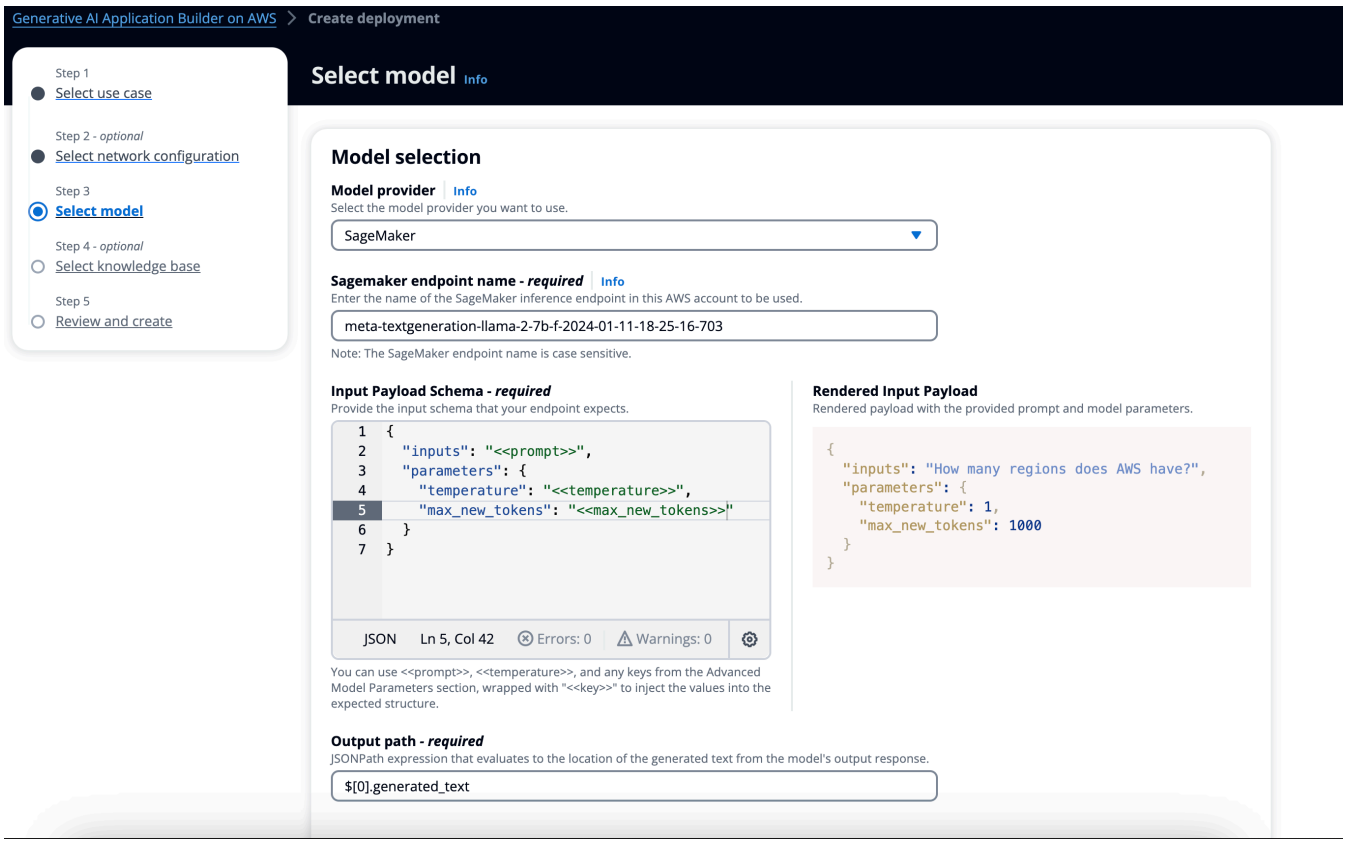
1. 透過部署儀表板精靈建立新的使用案例，並完成表單，直到到達模型選取頁面為止。
2. 在模型頁面上，選取 SageMaker AI 作為模型提供者。這將產生需要三個關鍵使用者輸入的自訂表單：
   - 您要使用的 SageMaker AI 端點名稱。DevOps 使用者可以從 AWS 主控台取得此資訊。請注意，端點必須位於部署解決方案所在的相同帳戶和區域中。

     AWS 主控台上端點名稱的位置



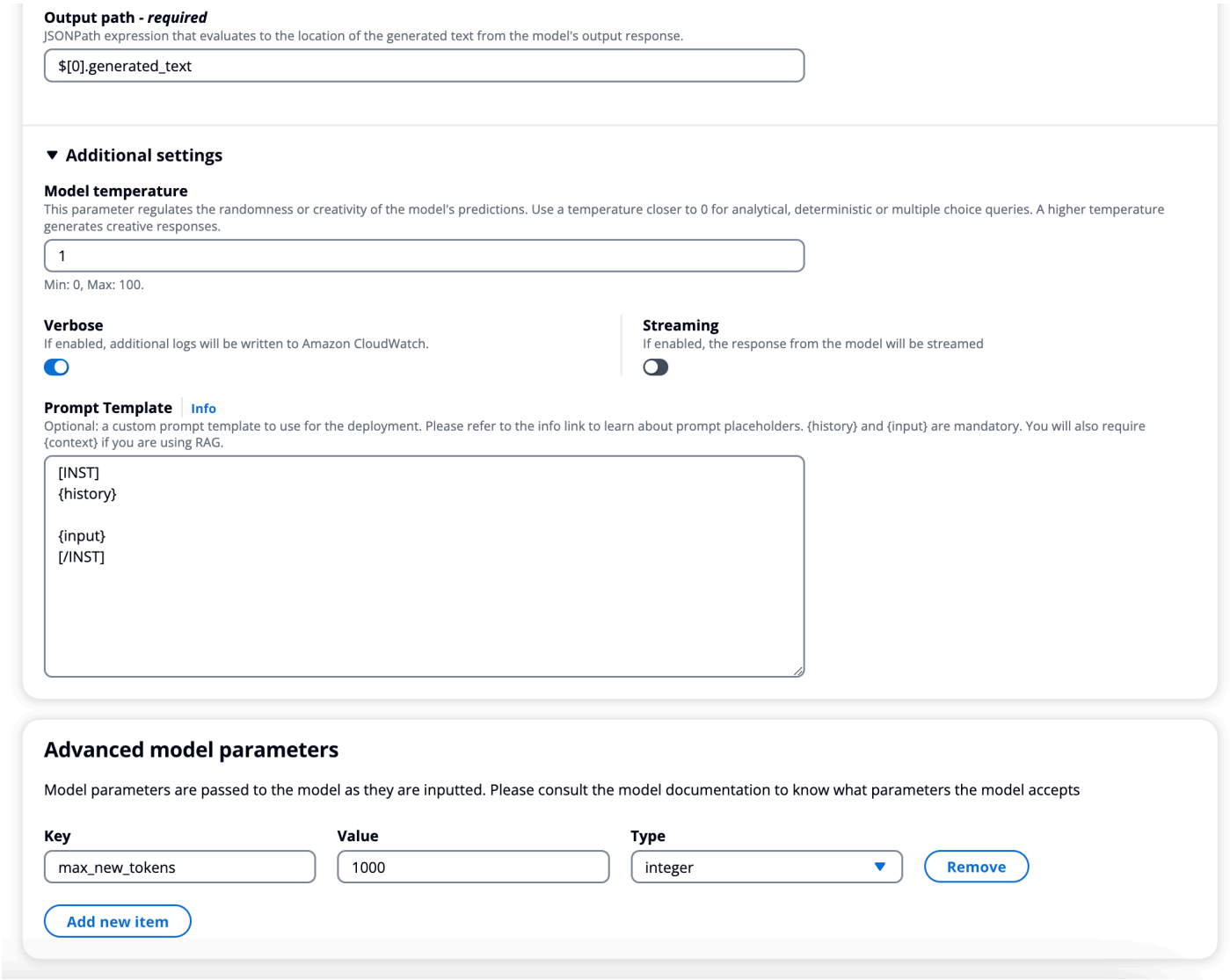
- 端點預期的輸入承載結構描述。若要支援最廣泛的端點集，管理員使用者必須告知解決方案其端點預期輸入格式化的方式。在模型選擇精靈中，提供要傳送至端點之解決方案的 JSON 結構描述。您可以新增預留位置，將靜態和動態值注入請求承載。可用選項為：
  - 強制預留位置：\<\<prompt\>\> 將動態取代為完整輸入（例如，歷史記錄、內容和根據提示範本的使用者輸入），以在執行時間傳送至 SageMaker AI 端點。
  - 選用預留位置：\<\<temp\>\> *，* 以及進階模型參數中定義的任何參數都可以提供給端點。任何包含以 \<\< 和 \>\> 括住預留位置的字串（例如，\<\<max_new_tokens\>\>) 都會被相同名稱的進階模型參數值取代。

    範例輸入結構描述 - 設定必要欄位、提示和溫度，以及自訂進階參數 max_new_tokens。輸出路徑必須以有效的 JSONPath 字串提供



3. LLMs 在輸出承載中產生字串回應的位置。這必須以 JSONPath 表達式形式提供，以指出向使用者顯示的最終文字回應預期從端點的傳回物件和回應中存取的位置。

   新增要在 SageMaker AI 輸入結構描述中使用的進階模型參數範例（請參閱圖 2 以取得先前的選項/設定）



> **Note**
>
> SageMaker AI 現在支援在相同端點後方託管多個模型，這是在目前版本的 SageMaker AI Studio（非 Studio Classic) 中部署端點時的預設組態。
> 如果您的端點以這種方式設定，您將需要將 InferenceComponentName 新增至進階模型參數區段，其值對應於您想要使用的模型名稱。

進階 LLM 設定

使用 Amazon Bedrock 時，您可以為模型設定一些進階設定，例如 Amazon Bedrock Guardrails、Amazon Bedrock 的佈建輸送量，以及其他模型參數。

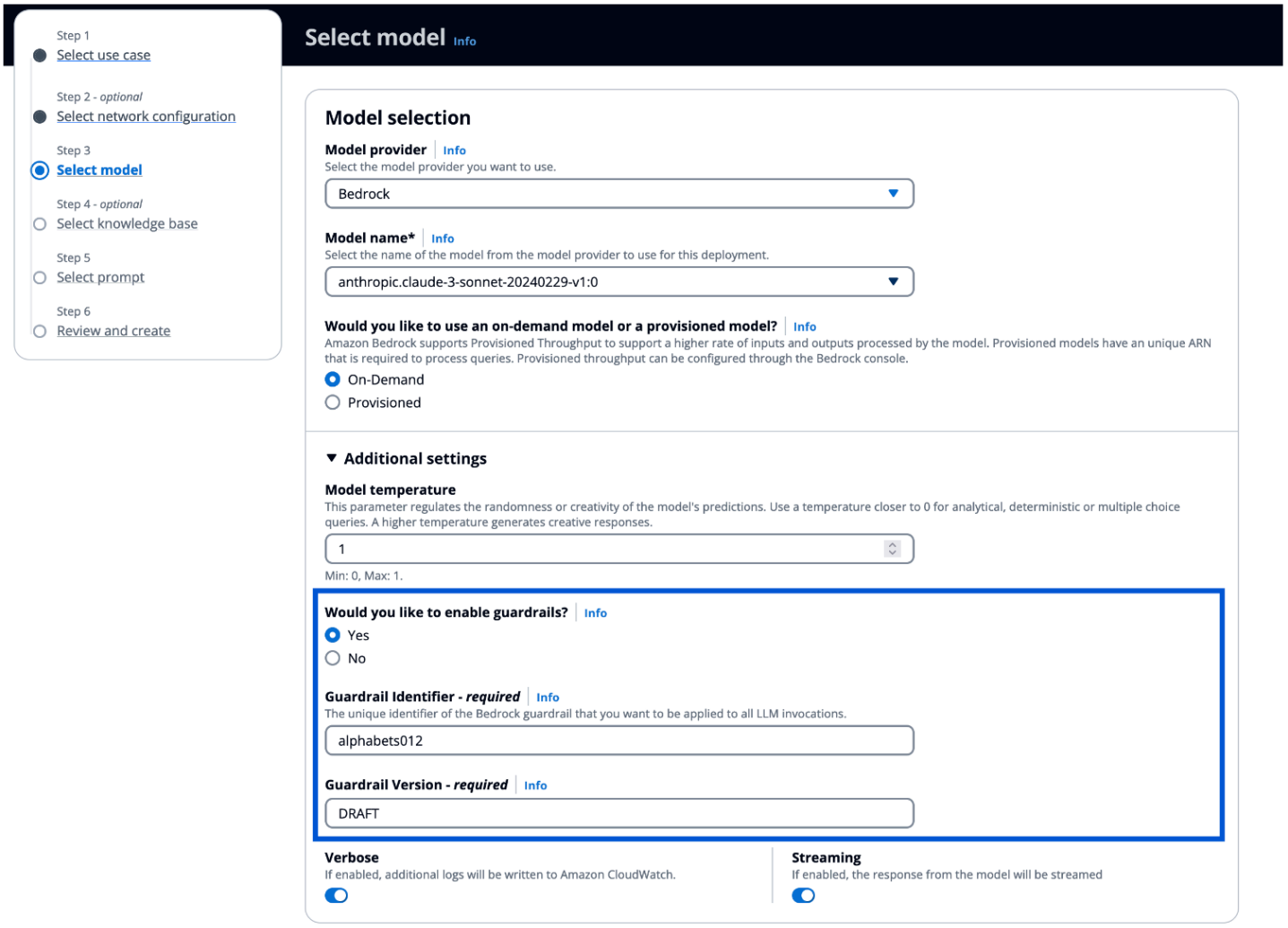
Amazon Bedrock 防護機制

Amazon Bedrock Guardrails 是 Amazon Bedrock 的一項功能，可根據使用者設定的政策評估使用者輸入和 LLM 回應，並提供額外一層保護，無論使用者為使用案例選擇的基礎 LLM 為何。護欄包含 2 個政策，可避免屬於不良或有害類別的內容：

1. 拒絕主題以定義一組在使用者應用程式內容中不理想的主題，例如金融應用程式的投資建議，以及
2. 內容篩選條件****允許篩選包含有害內容的輸入使用者提示或模型回應。

對於生成式 AI 應用程式建置器解決方案中的使用，必須使用建立護欄精靈在 Amazon Bedrock 主控台中設定護欄。建立後，您可以提供 Guardrail 識別符和 Guardrail 版本，將此 Guardrail 新增至模型選擇步驟中的其他設定中，透過生成式 AI Application Builder 解決方案精靈建立的聊天使用案例。

描述部署精靈 - 啟用 Amazon Bedrock 護欄

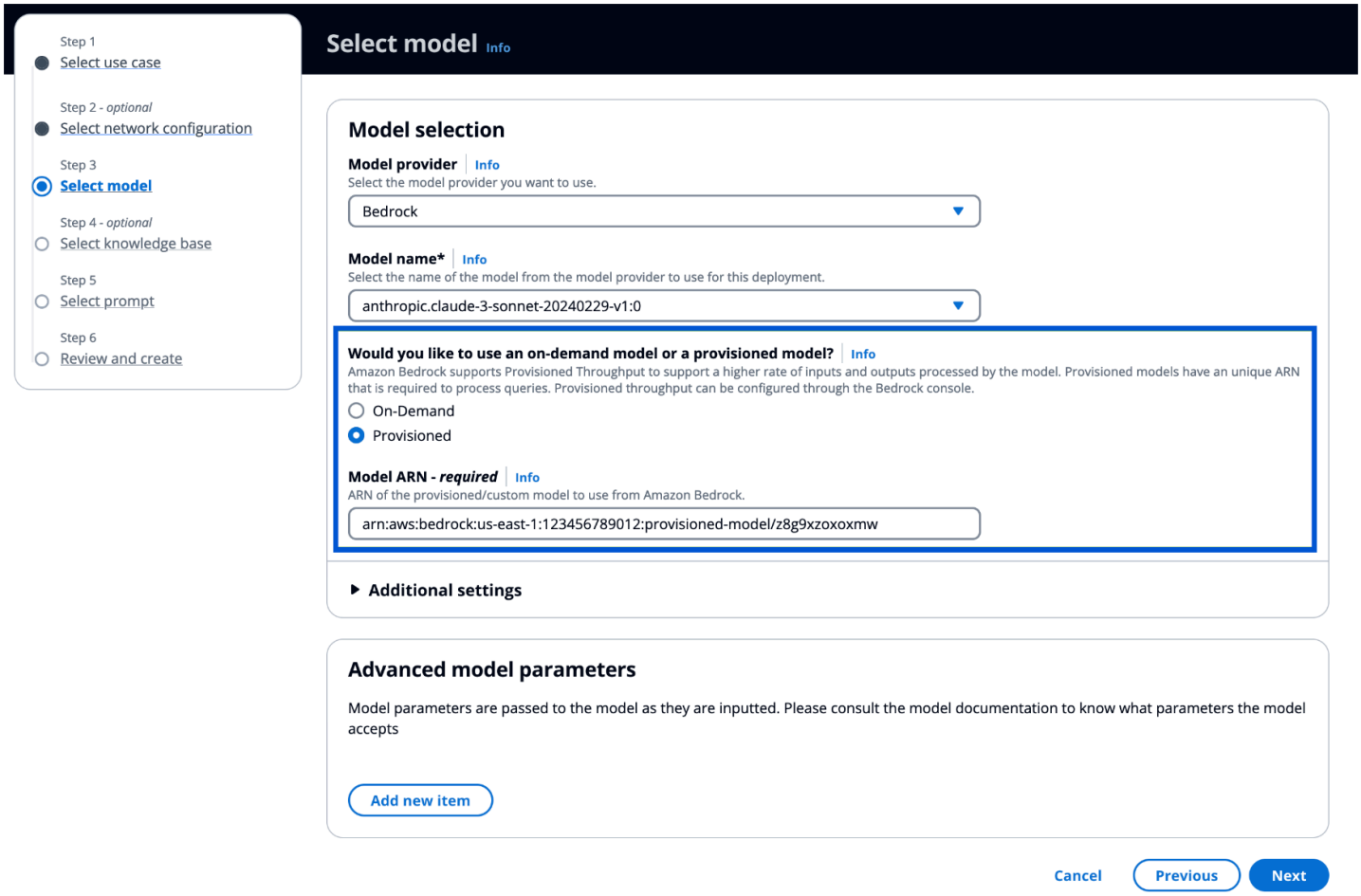


Amazon Bedrock 的佈建輸送量

每個隨需 Amazon Bedrock 模型都遵循特定區域的模型推論帳戶配額限制。例如，Batrock 上的 Anthropic Claude 2.x 目前允許在 us-east-1 和 us-west-2 區域中每分鐘處理 500 個請求和 500，000 個字符。您也可以將解決方案與經過微調或持續預先訓練的模型搭配使用。對於這類執行個體，Amazon Bedrock 允許佈建的輸送量，允許為您的基礎執行大型一致的推論工作負載、微調或持續預先訓練的模型，以用於生產級應用程式。

在 Amazon Bedrock 主控台中購買佈建輸送量後，就會產生模型 ARN 以供使用。您現在可以在模型選取步驟的生成式 AI 應用程式建置器精靈中提供此模型 ARN。若要這樣做，請選取 Bedrock 做為模型提供者，以及用來在 Amazon Bedrock 主控台中產生此佈建模型 ARN 的基本模型名稱。然後，選擇隨需模型和佈建模型時，選取「佈建模型」，並提供模型 ARN。

描述部署精靈 - 啟用 Amazon Bedrock 的佈建輸送量



> **Note**
>
> 您的護欄和佈建輸送量必須與部署的部署儀表板和使用案例堆疊位於相同的區域。

模型參數

LLMs通常接受其實作特有的各種參數。模型提供者通常會提供概述支援參數集及其用途的文件。

解決方案會將模型參數直接傳遞至基礎模型，因此請務必確保參數設定正確。如需支援參數的最新資訊，請參閱模型提供者的文件。

設定 代理程式建置器

Agent Builder 提供建立生產就緒 AI 代理器的完整組態選項。本節說明如何設定和管理 Agent Builder 部署。

系統提示組態

系統提示會定義代理程式的行為、人格和功能。若要設定系統提示：

1. 在客服人員建置器精靈中，導覽至設定客服人員步驟。
2. 在文字編輯器中編輯系統提示範本。
3. 包含以下項目的明確指示：
   - 客服人員的角色和目的
   - 如何使用可用的工具 (MCP 伺服器)
   - 回應格式偏好設定
   - 行為指導方針
4. 如有需要，請使用重設為預設按鈕來還原原始範本。

客服人員提示的最佳實務：

- 具體說明代理程式的功能和限制
- 提供所需行為的明確範例
- 包含工具使用方式和調用時機的指示
- 定義回應格式期望
- 設定客服人員行為的界限

MCP 伺服器整合

模型內容通訊協定 (MCP) 伺服器可讓代理程式存取企業工具和資料來源。若要設定 MCP 伺服器：

1. 在設定代理程式步驟中，找到 MCP 伺服器區段。
2. 從下拉式功能表中的可用 MCP 伺服器中選取 。

> **Note**
>
> 在部署代理程式之前，必須先設定和存取 MCP 伺服器。代理程式會自動探索和使用所設定 MCP 伺服器公開的工具。請參閱 MCP 文件以取得伺服器設定和工具組態。

記憶體設定

Agent Builder 提供兩種類型的記憶體來維護內容和知識：

短期記憶體

預設為所有客服人員啟用：

- 在工作階段中維護對話內容
- 自動擷取使用者訊息和客服人員回應
- 由 actorId 和 sessionId 組織，以進行適當的隔離
- 不需要組態

長期記憶體

用於跨工作階段存放洞見的選用功能：

1. 在設定代理程式步驟中，找到記憶體組態區段。
2. 切換 啟用要啟用的長期記憶體。
3. 啟用時，代理程式可以：
   - 跨對話擷取和存放重要資訊
   - 從先前的工作階段擷取相關內容
   - 建立使用者偏好設定和歷史記錄的相關知識

> **Note**
>
> 長期記憶體使用 AgentCore 記憶體搭配語意記憶體策略和預設保留設定。

監控 代理程式建置器部署

Agent Builder 透過 CloudWatch 儀表板和指標提供全面的監控。

存取 CloudWatch 儀表板

1. 導覽至 AWS 帳戶中的 CloudWatch 主控台。

2. 從左側導覽選取儀表板。
3. 尋找名為 的儀表板AgentBuilder-<UseCaseId>。
4. 檢視即時指標和歷史效能資料。

日誌存取和分析

代理程式日誌可在 CloudWatch Logs 中取得：

1. 在 AWS 主控台中導覽至 CloudWatch Logs。
2. 尋找字首為 的日誌群組/aws/bedrock-agentcore/runtimes/。
3. 使用 CloudWatch Insights 查詢和分析日誌。
4. 搜尋特定請求 IDs或錯誤模式。

設定工作流程建置器

Workflow Builder 透過將工作委派給專業客服人員建置器客服人員的主管客服人員，啟用多重客服人員協調。

建立工作流程

1. 導覽至部署儀表板
2. 選取建立工作流程使用案例
3. 設定主管代理程式：
   - 名稱：工作流程的描述性名稱
   - 描述：目的和功能
   - 系統提示：客服人員委派和協調的指示
   - 模型：主管客服人員的基礎模型

主管提示的最佳實務：

- 清楚描述何時使用每個專用代理程式
- 包含從多個客服人員彙總結果的指示
- 定義回應格式設定期望
- 設定委派行為的界限

客服人員選擇

選取客服人員建置器客服人員，以包含為專門客服人員：

1. 按一下工作流程組態中的新增代理程式
2. 瀏覽或搜尋可用的客服人員建置器客服人員
3. 檢閱客服人員描述
4. 選取要包含在工作流程中的客服人員

客服人員描述

主管客服人員使用客服人員描述來決定要委派給哪個客服人員。確保描述清楚說明：

- 客服人員的專業網域或功能
- 代理程式處理的任務類型
- 輸入/輸出期望

測試工作流程

部署之後：

1. 透過部署儀表板存取工作流程
2. 使用需要多個代理程式的查詢進行測試
3. 在 CloudWatch 日誌中監控代理程式委派
4. 檢閱回應品質和委派模式
5. 如果委派不理想，請調整主管提示

管理模型字符限制的提示

注意：解決方案不會直接嘗試管理各種 LLMs字符限制。測試並確保您的提示保持在模型提供者強制執行的可用限制內。

若要協助控制提示的大小，請嘗試下列動作：

1. 熟悉您想要使用的模型所施加的限制。這些值在各個模型之間可能會有很大的差異，因此在開始之前，請務必了解可用的預算。

2. 建立您的初始提示時，請謹記該預算，並考慮要為提示的任何動態元素節省多少成本。例如，使用者輸入、聊天歷史記錄、文件摘錄等。
3. 在提示組態頁面中，設定追蹤歷史記錄的大小限制，以限制提示中包含的對話轉彎次數。
4. 在知識庫組態精靈中設定文件傳回限制。您需要嘗試在為 LLM 提供足夠的內容來執行任務之間取得正確的平衡，但不能超過字符限制或對延遲造成負面影響。
5. 保留一些緩衝區。不要編列典型案例的預算，請考慮並實驗邊緣案例，例如長輸入查詢、大型文件摘錄或長對話。

建置 MCP 伺服器 Docker 映像的步驟

若要在 AWS 上使用 MCP （模型內容通訊協定） 伺服器與生成式 AI 應用程式建置器，您需要在私有 Amazon ECR 儲存庫中建置和存放的 Docker 映像作為第一步。

> **Note**
>
> 截至目前為止，Amazon Bedrock AgentCore 執行時間中現有的部署 MCP 伺服器無法匯出至 GAAB。若要將 MCP 伺服器連接到透過 GAAB 建立的代理程式，他們需要透過 GAAB 建立。

步驟 1：建立 MCP 伺服器

首先，您需要準備好 MCP 伺服器實作。如需建立 MCP 伺服器的詳細指示，請參閱 Amazon Bedrock AgentCore 開發人員指南 - 建立 MCP 伺服器。

我們建議使用下列專案結構：

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

對於 Dockerfile 結構，建議使用類似下列範例的格式：

```
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"]
```

步驟 2：在本機測試 MCP 伺服器

在部署到 AWS 之前，請務必在本機測試 MCP 伺服器，以確保其如預期般運作。如需本機測試的詳細說明，請參閱 Amazon Bedrock AgentCore 開發人員指南 - 在本機測試 MCP 伺服器。

步驟 3：部署至 Amazon ECR

在本機建立和測試 MCP 伺服器後，請依照下列步驟將其部署至 Amazon ECR：

1. 請確定您已安裝最新版本的 AWS CLI 和 Docker。如需詳細資訊，請參閱 Amazon ECR 入門。

2. 擷取身分驗證字符，並向登錄檔驗證 Docker 用戶端。使用 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. 使用下列命令建置 Docker 映像。如需從頭開始建置 Docker 檔案的資訊，請參閱 Docker 文件。如果您的映像已建置完成，您可以略過此步驟：

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

4. 建置完成後，請標記您的映像，以便您可以將映像推送到此儲存庫：

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

5. 執行下列命令，將此映像推送至您新建立的 AWS 儲存庫：

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

如需完整的部署說明，請參閱 Amazon Bedrock AgentCore 開發人員指南 - 將 MCP 伺服器部署至 AWS。

步驟 4：在 GAAB 中使用 ECR URI

成功將 Docker 映像推送至 Amazon ECR 之後，請從 ECR 主控台複製映像 URI。透過 AWS 部署精靈上的生成式 AI 應用程式建置器部署 MCP 伺服器時，您將使用此 URI。

建立不同 MCP 閘道目標的步驟

Amazon Bedrock AgentCore Gateway 可讓您將現有的 AWS 服務和 APIs 轉換為可供您的代理程式使用的 MCP 工具。Gateway 支援多種目標類型，可讓您無縫整合各種後端服務。

支援下列目標類型：

- Lambda 目標：將 AWS Lambda 函數轉換為 MCP 工具。如需詳細說明，請參閱 Amazon Bedrock AgentCore 開發人員指南 - 新增 Lambda 目標。
- OpenAPI 目標：使用 OpenAPI 規格將 REST APIs定義為 MCP 工具並公開。如需詳細說明，請參閱 Amazon Bedrock AgentCore 開發人員指南 - OpenAPI 結構描述。

- Smithy 目標：使用 Smithy 模型定義建置 MCP 工具，以進行類型安全 API 整合。如需詳細說明，請參閱 Amazon Bedrock AgentCore 開發人員指南 - 建置 Smithy 目標。
- MCP 伺服器目標：透過 URL 端點直接連線至外部 MCP 伺服器，讓您整合現有的 MCP 伺服器。如需詳細說明，請參閱 Amazon Bedrock AgentCore 開發人員指南 - MCP 伺服器目標。

如需建立 MCP Gateway 目標的其他範例和教學課程，請造訪 Amazon Bedrock AgentCore 範例儲存庫。

設定知識庫

本節說明如何將資料擷取到您為解決方案選取的知識庫。解決方案目前支援 Amazon Kendra 和 Amazon Bedrock 知識庫做為以 RAG 為基礎的使用案例部署的知識庫。

Amazon Kendra

如果您使用 Amazon Kendra 做為知識庫，請參閱 Amazon Kendra 開發人員指南，了解如何使用各種資料來源連接器來協助您從廣泛的選擇來源擷取資料。

重要：為防止意外遺失資料，在刪除部署或堆疊時，解決方案不會自動刪除 Kendra 索引（無論是由解決方案或其他方式建立）。如果您想要刪除知識庫並停止產生成本，請參閱手動解除安裝一節，了解保留哪些資源以及如何清除這些資源的詳細資訊。

Amazon Bedrock 知識庫

Amazon Bedrock 知識庫可以由各種不同的向量存放區提供支援，每個存放區都具有編製資料索引的功能。若要設定和填入您的知識庫，請參閱 Amazon Bedrock 使用者指南。具體而言，您會想要：

- 首先設定您的資料來源
- 然後在支援的向量存放區中為您的知識庫設定向量索引。請注意，如果您在建立知識庫期間，在 Bedrock 主控台中使用「快速建立新的向量存放區」選項，則可以略過此選項。
- 最後，您可以建立知識庫並同步您設定的資料來源。

進階知識庫設定

進階知識庫設定，例如知識庫篩選和具有角色型存取控制的 RAG，可用於 解決方案。當具有角色型存取控制的 RAG 特別適用於 Amazon Kendra 時，知識庫篩選可以套用到其中一個知識庫。

知識庫篩選

解決方案可讓您在精靈知識庫步驟的進階 RAG 組態區段中部署使用案例時，指定 Amazon Kendra 屬性篩選條件或 Bedrock 知識庫擷取篩選條件。 https://docs.aws.amazon.com/bedrock/latest/userguide/kb-test-config.html這些篩選條件會定義如何查詢知識庫中的資料來源，例如搜尋策略、基礎文件的語言為查詢等。

在這兩種情況下，JSON 物件都會用來根據每個服務文件中指定的格式來指定篩選條件設定 （如上連結）。

範例 1：Kendra AttributeFilter

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

範例 2：Batrock RetrievalFilter

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

使用 Amazon Kendra 的角色型存取控制的 RAG

角色型存取控制 (RBAC) 允許控制哪些使用者或群組可以存取 Amazon Kendra 索引中的特定文件，或在搜尋結果中查看特定文件。若要使用 AWS (GAAB) 上的生成式 AI 應用程式建置器使用案例來設定 Amazon Kendra 索引 ID 的 RBAC，請遵循下列步驟：

1。設定 Amazon Kendra 索引

1. 確保您已建立 Amazon Kendra 索引，並至少新增一個資料來源。

2. 根據使用者群組設定資料來源的存取控制。對於 S3 資料來源，請遵循文件中的指示，使用在 Amazon Cognito 使用者集區中建立的相同群組名稱來設定存取控制清單 (ACLs)。這可確保使用者只能存取他們根據群組成員資格獲授權檢視的文件和搜尋結果。

> **Note**
>
> 在您建立的 Kendra 索引中的使用者存取控制下，將字符型使用者存取控制保留為否。 當您在步驟 2 中啟用角色型存取控制時，AWS 上的生成式 AI 應用程式建置器會從使用者身分驗證字符擷取適當的宣告，並建立屬性篩選條件。

2. 使用 GAAB 部署精靈部署 RAG 使用案例

1. 遵循 GAAB 部署精靈中的螢幕精靈說明，直到您達到精靈的步驟 4 來設定 RAG。
2. 在部署精靈的選取知識庫步驟中，選擇 Amazon Kendra 作為知識庫類型。
3. 指定您是否擁有現有的 Amazon Kendra 索引，還是要建立新的索引。如果您有現有的索引，請提供已根據使用者群組使用存取控制清單 (ACLs) 設定的 Amazon Kendra 索引 ID。
4. 啟用角色型存取控制選項。此選項可確保根據使用者的角色和群組許可，篩選從 Amazon Kendra 索引傳回的搜尋結果。
5. 檢閱和部署使用案例。

3. 設定 Amazon Cognito

1. 找出 GAAB 部署所使用的 Amazon Cognito 使用者集區。此 Amazon Cognito 使用者集區通常由主要部署儀表板 CloudFormation 堆疊建立。
2. 在 Amazon Cognito 使用者集區中建立新使用者。建立使用者時，請選取「傳送電子郵件邀請」選項，讓使用者透過電子郵件接收臨時登入憑證。這可讓新使用者註冊和存取 GAAB 應用程式。
3. 在 Amazon Cognito 使用者集區中建立使用者群組。確保群組名稱完全符合 Amazon Kendra 索引 ACLs 中設定的群組。這對於啟用 RBAC 至關重要，因為使用者的群組成員資格將決定他們可以存取的搜尋結果。
4. 根據使用者的角色和存取許可，將使用者指派給適當的群組。必須將使用者新增至 Amazon Kendra 索引 ACL 所需的群組，以及在 GAAB 部署期間建立的使用案例特定群組。這可確保使用者擁有存取特定使用案例和相關搜尋結果的必要許可。

遵循這些步驟，您將為您的 GAAB 部署設定角色型存取控制 (RBAC)，確保使用者只能根據其指派的使用者群組和許可，存取並與其授權的資訊和功能互動。

> **Note**
>
> 到目前為止，只有 Amazon Kendra 在 AWS 上的生成式 AI 應用程式建置器中支援知識庫的 RBAC。對於 Amazon Bedrock 知識庫，不支援 RBAC，但您可以使用中繼資料篩選條件來實現某種程度的篩選。如需詳細資訊，請參閱 Amazon Bedrock 使用者指南。

設定您的提示

部署儀表板精靈具有提示組態步驟，可讓您自訂提示體驗和範本，以引導使用者和 AI 模型之間的互動。正確設定這些設定對於從 AI 助理取得準確且相關的回應至關重要。

本節控制 AI 提示的整體體驗和行為。

- 提示範本長度上限：此設定決定提示範本的長度上限（以字元為單位）。較高的值可讓 AI 模型提供更多內容，進而產生更準確的回應。不過，過長的提示也可能會產生雜訊，並對效能產生負面影響。對於 Amazon Bedrock 模型，使用基礎模型字符限制計算提示範本長度上限的預設值（以字元為單位）。如果您在 Bedrock 中編輯和變更模型名稱，「重設為預設值」按鈕會反白顯示，並可用來採用新選取的模型預設值。對於 Amazon SageMaker AI 模型，提供了合理的預設值，但建議您檢查基礎模型，並相應地選擇這些提示範本長度上限和輸入文字長度。如需詳細資訊，請參閱管理模型字符限制的秘訣一節。
- 輸入文字長度上限：此設定會限制使用者輸入文字的長度上限（以字元為單位）。較長的輸入可能包含不相關的資訊，增加從 AI 模型取得不相關或不準確回應的風險。
- 使用者提示編輯：此選項可讓您啟用或停用使用者透過聊天 UI 修改提示範本的功能。停用此功能有助於維持一致性，並防止對提示進行意外變更。

提示範本

本節可讓您定義 AI 模型將使用的實際提示範本。提示範本通常會遵循包含各種元件預留位置的結構，例如使用者的輸入、參考段落和聊天歷史記錄。

- 提示範本：這是您可以撰寫或貼上所需提示範本的主要文字區域。範本應製作為 AI 模型提供必要的內容和指示。它通常包含下列預留位置：
  - `{input}`：此預留位置對於 Sagemaker AI 部署而言是強制性的，並將取代為使用者的輸入或查詢。

- `{history}`：此預留位置對於 Sagemaker AI 部署而言是強制性的，並將取代為目前對話的聊天歷史記錄。
- `{context}`：此預留位置對於 RAG 部署是強制性的，並將取代為從設定的知識庫取得的文件摘錄。

- 重述問題？：此選項（僅適用於 RAG 部署）會決定在傳遞給 AI 模型之前，是否應重述或取消歧義使用者的原始輸入查詢。重新表達查詢有時有助於模型更了解使用者的意圖，進而產生更準確的回應。

設定提示範本和體驗時，請務必在為 AI 模型提供足夠的內容和指示之間取得平衡，同時避免過於長或無關的資訊，這可能會導致雜訊或效能問題。

進階提示設定

本節可讓您控制如何向 AI 模型呈現對話歷史記錄。

- 追蹤歷史記錄的大小：此設定會決定應該包含在最終提示中的先前訊息數量。將此值設為零會導致沒有歷史記錄插入提示範本或歧義提示範本。請注意：即使設為零，仍然需要在提示範本中存在 {history} 預留位置。在執行時間，它會取代為空字串。
  - 注意：建議為此值提供偶數。提供奇數只會傳回配對互動的 AI 回應。
- 人力字首：這是用來識別使用者在對話歷史記錄中傳送之訊息的字首。
- AI 字首：這是用來識別對話歷史記錄中 AI 模型傳回之訊息的字首。

歧義提示詞組態

本節可讓您在將使用者輸入傳送至設定的知識庫之前，設定用於取消混淆使用者輸入的行為和範本。

- 啟用歧義：此選項會判斷使用者輸入是否應該在傳送至知識庫之前歧義。
- 歧義提示範本：這是提示範本，用於在連線到知識庫時歧義使用者輸入。從此提示產生的輸出將用作傳送至知識庫的查詢。停用歧義會導致使用者的原始查詢未變更地傳送至知識庫。

例如，啟用歧義時，後續使用者查詢「費用是多少？」 可能會混淆為「續約我的車牌的費用是多少？」，進而產生更好的搜尋查詢。

使用部署的文字使用案例

文字使用案例的內建 UI 旨在讓商業使用者快速探索和實驗管理員使用者建立的部署。商業使用者所做的組態變更只會對其工作階段生效。商業使用者必須與管理員使用者共用這些變更，管理員使用者可以使用這些變更來更新基本部署，以供所有 使用。

聊天 UI 包含下列元件：

- 聊天視窗
- 聊天輸入方塊
- 設定
- 清除對話

聊天視窗

保留不同的對話轉彎。從右側開始的訊息來自商業使用者，從左側開始的訊息來自設定的 LLM。所有 LLM 回應上都存在一個小型剪貼簿圖示，以便輕鬆複製回應。

聊天輸入方塊

固定於聊天視窗底部是聊天輸入方塊。在這裡，商業使用者可以輸入要傳送到 LLM 的訊息。輸入方塊正上方是連線狀態。如果連線遺失（例如，由於閒置），下次傳送聊天訊息時會自動建立新的連線。由於額外的 WebSocket 連線時間，此請求預計需要更長的時間。

根據特定組態，輸入可能會強制執行長度上限。如果超過此限制，使用者會收到提醒，而且不會傳送訊息。

注意：如果搭配 Amazon Kendra 使用 RAG，則擷取 API 會將查詢截斷為 30 個字符字。如果預期使用者輸入時間較長，請評估這可能會影響搜尋效能。

設定

為了讓商業使用者能夠快速試驗不同的組態，可使用設定面板，以on-the-fly編輯特定部署組態選項

（例如提示範本）。這些變更只能在新工作階段開始時進行。對話開始後，清除對話會重新啟用組態設定的編輯。

注意：管理員使用者可以選擇鎖定部署的設定。在提示步驟期間，他們可以防止透過精靈在部署時間進行即時編輯。

清除對話

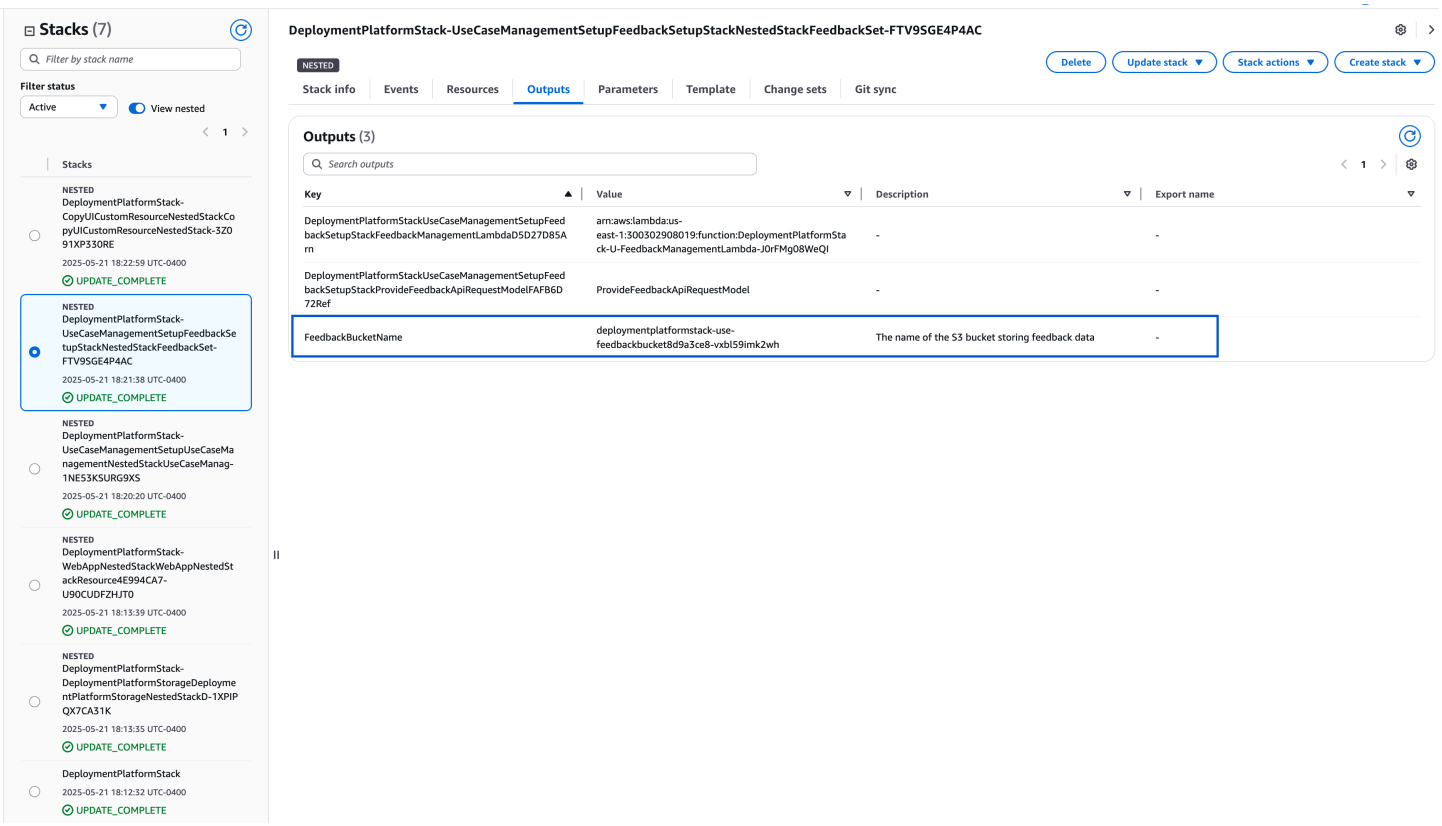
在對話過程中，解決方案會維護聊天歷史記錄，以啟用對話體驗。這可啟用查詢歧義和後續問題。若要重設對話並刪除此互動的所有聊天歷史記錄，請選擇聊天視窗頂端的 *清除對話 *。對話清除後，會建立新的工作階段，以重新啟用編輯設定。

存取和分析使用者收集的意見回饋

自 v3.0.0 起，部署儀表板會部署巢狀意見回饋堆疊，允許使用儀表板部署的文字和 Bedrock 代理程式使用案例，具有 LLM/代理程式產生之回應的意見回饋集合功能。特別是，使用者可以提供正面或負面的意見回饋，以及選用的評論。如果使用者提供負面意見回饋，他們可以進一步選取其中一個負面類別：「不正確」、「不完整或不足」、「有害」和/或「其他」。

使用者提供意見回饋後，意見回饋會存放在依使用案例 ID、年份和月份分割的 S3 儲存貯體中。您可以在部署儀表板中找到使用案例 ID，也可以在部署儀表板堆疊的回饋巢狀堆疊的輸出中找到回饋 S3 儲存貯體：

描述部署堆疊 - 尋找意見回饋儲存貯體名稱



使用者意見回饋會以包含最少一組資訊的 API 請求傳送：

```
{
  "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."
}
```

然後，Lambda 會使用 在部署時識別使用案例正確組態的 useCaseRecordKey 來處理此承載。此組態用於取得意見回饋的特定詳細資訊，例如 ConversationTable 的名稱（包含所有對話和人工和 AI 訊息序列），而這些項目進一步用於擷取實際 userInput和 llmResponse。其他詳細資訊也會附加到此意見回饋記錄，例如 Bedrock 代理程式使用案例agentAliasId的 agentId和 modelProvider，以及使用此組態的文字使用案例的 bedrockModelId、 等。如需如何存取此組態的詳細資訊，請參閱下面的自訂意見回饋映射一節。每個傳入的意見回饋請求都會儲存為 JSON 物件，而範例意見回饋記錄看起來像文字使用案例：

```
{
   "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"
}
```

對於 Bedrock 代理程式使用案例，或類似項目：

```
{
   "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"
}
```

然後，此意見回饋可用於進一步處理、分析和建立重新訓練/意見回饋迴圈的模型。您也可以新增自訂映射，以增強儲存在意見回饋 lambda 中的意見回饋記錄。

自訂意見回饋映射

部署儀表板包含的 LLMConfigTable可在具有金鑰 的部署儀表板堆疊的堆疊輸出中找到LLMConfigTableName。 LLMConfigTable 包含每個使用案例的組態，根據管理員在透過部署儀

表板精靈部署使用案例時選取的設定。每個使用案例組態都由其 識別useCaseRecordKey。以下是中的範例使用案例組態記錄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"
    }
}
```

如果已啟用使用案例的意見回饋，此組態將包含FeedbackParams一個物件，允許其內部的CustomMappings物件指定要新增至意見回饋 S3 儲存貯體中存放的意見回饋 JSON 記錄之所有其他欄位的 JSONPaths。例如，對於上述範例使用案例組態，CustomMappings 還會在以 config作為 ScoreThreshold JSONPaths 根目錄的 CustomMappings 物件中包含 NumberOfDocs和 JSONPaths。透過此組態，儲存在意見回饋 S3 儲存貯體中的每個 JSON 記錄將開始從已提供的欄位取得這 2 個額外值。

分析意見回饋資料

意見回饋資料會以 JSON 物件的形式存放在 S3 中。以下是讓此意見回饋資料更易於存取和可行的一些方法：

使用 AWS Glue 和 Amazon Athena

AWS Glue 和 Amazon Athena 提供無伺服器方法來編目、查詢和分析您的意見回饋資料。

AWS Glue 可讓您建立 AWS Glue 爬蟲程式，以檢查 S3 儲存貯體中的資料、推斷其結構描述，並記錄目錄中所有相關中繼資料。之後，Amazon Athena 之類的服務可用於查詢資料。

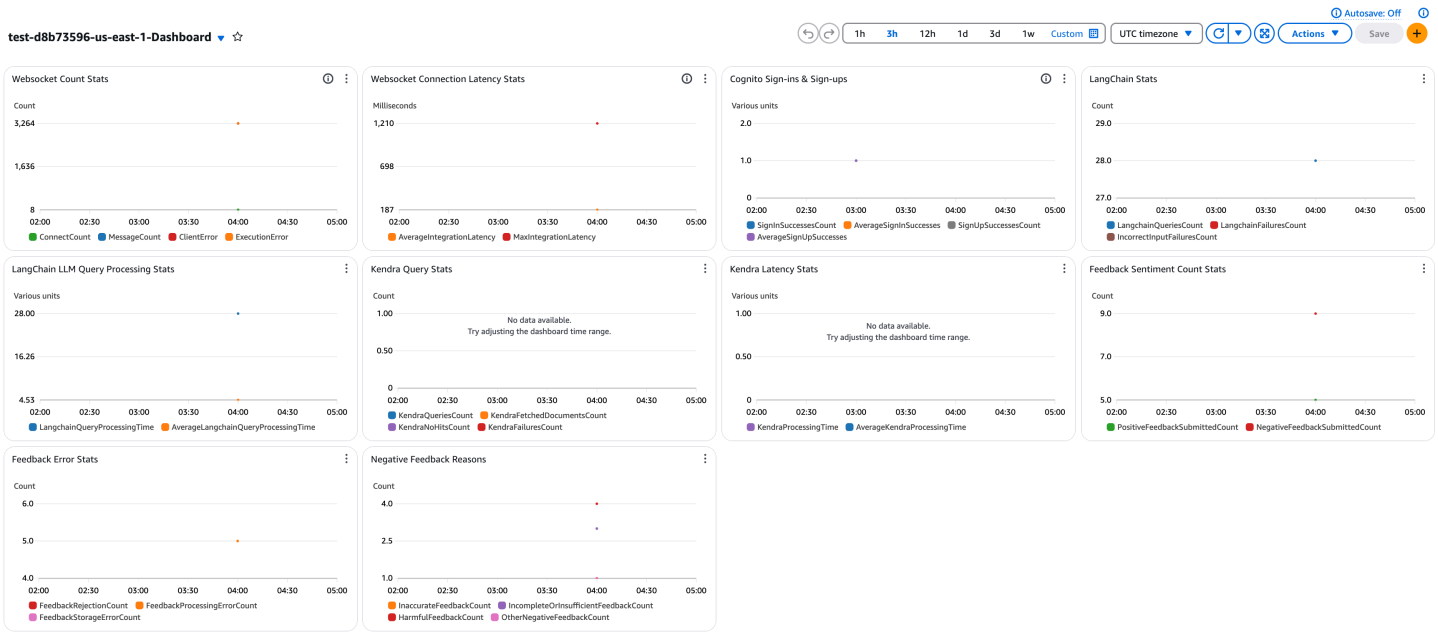
您可以參考 AWS Athena 文件，了解使用 AWS Glue Data Catalog 將意見回饋 S3 儲存貯體與 Amazon Athena 連線的步驟。您也可以使用 Glue 一些更強大的功能，對此資料執行擷取轉換和載入 (ETL) 任務，並將其轉換為適合您分析或模型重新訓練使用案例的格式。使用 Glue，您可以執行一些操作，例如篩選具有特定意見回饋類型的記錄、填寫任何缺少的資訊，也可以將此資料載入另一個儲存位置，例如另一個 S3 儲存貯體或不同的 AWS 資料存放區。

> **Note**
>
> 根據您的使用案例，請考慮排定 Glue 爬蟲程式定期執行（例如每週）而不是每天執行，以最佳化成本，因為意見回饋資料可能很稀疏。

使用解決方案的 CloudWatch Dashboards

您也可以存取與 解決方案一起封裝的 CloudWatch Dashboard，以根據每個使用案例為您提供正面和負面意見回饋、負面意見回饋原因類別等趨勢。您可以在 AWS CloudWatch 主控台的 Dashboards 中，使用您的使用案例名稱來尋找此儀表板：

描述使用案例 CloudWatch 儀表板



您也可以在此儀表板中建置其他小工具，或建立 Amazon Quick Sight 儀表板。

意見回饋資料分析的最佳實務

- 在 S3 儲存貯體上實作資料生命週期政策，將較舊的意見回饋資料封存至成本較低的儲存層
- 為每個使用案例建立個別分析，以識別特定模型的改善機會
- 建立意見回饋閾值，在負面意見回饋超過可接受的層級時觸發提醒
- 定期匯出重要洞見，以便與利益相關者和模型改進團隊共用

檢視部署的操作指標

部署儀表板和使用案例堆疊各隨附自己的 CloudWatch 儀表板，可追蹤解決方案的各種操作指標。您可以使用這些 CloudWatch 儀表板來協助比較不同的部署。若要存取儀表板：

1. 導覽至 CloudWatch 主控台。
2. 透過查詢堆疊名稱或通用唯一識別符 (UUID) 來搜尋預先建置的儀表板。

例如，文字使用案例隨附圖形，可追蹤 WebSocket 連線的數量、使用者登入和註冊的數量、LLM 處理完成所花費的時間，以此類推。客戶可以使用這些圖表來比較部署的各種 _quantitative _metrics。

Example

很難比較套用至不同使用案例的各種模型的定性結果。使用複製功能快速啟動多個部署，以便您可以並排比較輸出。

存取 CloudWatch Logs 洞察

此解決方案會記錄 Lambda 函數的錯誤、警告、資訊和偵錯訊息。若要選擇要記錄的訊息類型：

1. 在 AWS Lambda 主控台中尋找適用的函數。
2. 新增 POWERTOOLS_LOG_LEVEL 環境變數。
3. 將變數設定為適用的訊息類型。

如需進一步指示，請參閱AWS Lambda Lambda 開發人員指南》中的建立 Lambda 環境變數。

下表列出您可以選擇的日誌層級類型。

Level	Description
ERROR (錯誤)	日誌包含任何導致 操作失敗的資訊。
WARNING	日誌包含任何可能導致函數不一致，但不一定會導致操作失敗的資訊。日誌也包含錯誤訊息。
INFO	日誌包含有關函數運作方式的高階資訊。日誌也包含 ERROR 和 WARNING 訊息。
除錯	日誌包含偵錯函數問題時可能有幫助的資訊。日誌也包含 ERROR、WARNING 和 INFO 訊息。

使用下列程序將 CloudWatch Logs 洞察新增至此解決方案。

1. 識別相關的日誌群組：
   a. 登入 AWS CloudFormation 主控台。
   b. 選擇您的目標堆疊。
   c. 選取資源索引標籤，並搜尋您的目標 Lambda 函數。
   d. 登入 AWS Lambda 主控台，然後選擇每個目標 Lambda 函數。

e. 針對每個目標 Lambda 函數，選取監控索引標籤，然後選擇檢視 CloudWatch Logs。

f. 複製您要從中擷取洞見的日誌群組名稱。

2. 導覽至 Amazon CloudWatch 主控台。

3. 在導覽功能表的日誌下，選擇日誌洞見。

4. 在 Logs Insights 頁面上，選擇 Logs 索引標籤。

5. 從步驟 1 搜尋日誌群組名稱。

6. 複製下列其中一個範例查詢，並將其貼到查詢欄位中：

a. 若要識別所有用戶端例外狀況：

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

b. 若要依函數名稱擷取調用計數：

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

c. 若要擷取五分鐘間隔內的調用計數：

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

d. 若要擷取所有 AWS X-Ray IDs：

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

e. 若要擷取與特定 X-Ray 追蹤 ID 相關的日誌：

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

f. 若要擷取未經授權的 WebSocket 錯誤：

```
fields
@ingestionTime,
@log,
@logStream,
@message,
@requestId,
@timestamp,
errorMessage,
```

```
errorType
|filter @message like /Unauthorized/ and @message like /websocket/|sort @timestamp
 desc
```

g. 若要擷取已發佈的指標計數：

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

開發人員指南

本節提供解決方案的原始程式碼、整合指南、自訂指南和 API 參考。

來源碼

請造訪我們的 GitHub 儲存庫，下載此解決方案的來源檔案，並與他人共用您的自訂項目。

AWS 範本上的生成式 AI 應用程式建置器是使用 AWS 雲端開發套件 (AWS CDK) 產生。如需其他資訊，請參閱 README.md 檔案。

整合指南

整個解決方案的設計可輕鬆擴充。此解決方案的協同運作層是使用 LangChain 建置。您可以將 LangChain （或為這些元件提供 LangChain 連接器的第三方） 支援的任何模型提供者、知識庫或對話記憶體類型新增至此解決方案。

展開支援的 LLMs

若要新增另一個模型提供者，例如自訂 LLM 提供者，您必須更新解決方案的下列三個元件：

1. 建立新的 `TextUseCase` CDK 堆疊，部署使用自訂 LLM 提供者設定的聊天應用程式：
   a. 複製此解決方案的 GitHub 儲存庫，並遵循 README.md 檔案中提供的指示來設定您的建置環境。
   b. 複製 （或建立新的） `source/infrastructure/lib/bedrock-chat-stack.ts` 檔案，將其貼到相同的目錄，並將其重新命名為 `custom-chat-stack.ts`。
   c. 將 檔案中的 類別重新命名為適當的類別，例如 `CustomLLMChat`。
   d. 您可以選擇將 Secrets Manager 秘密新增至此堆疊，以存放自訂 LLM 的登入資料。您可以在下一段討論的聊天 Lambda 層中，在模型調用期間擷取這些登入資料。
2. 建置並連接包含要新增之模型提供者 Python 程式庫的 Lambda 層。對於 Amazon Bedrock 使用案例聊天應用程式，`langchain-aws`Python 程式庫包含 LangChain 套件上方的自訂連接器，以連接至 AWS 模型提供者 (Amazon Bedrock 和 SageMaker AI)、知識庫 (Amazon Kendra 和 Amazon Bedrock 知識庫） 和記憶體類型 （例如 DynamoDB)。同樣地，其他模型提供者也有自己的連接器。此層可協助您連接此模型提供者的 Python 程式庫，以便在聊天 Lambda 層中使用這些連接器，這會叫用 LLM （步驟 3)。在此解決方案中，自訂資產綁定器用於建置使用 CDK 層面連接的 Lambda 層。若要為自訂模型提供者程式庫建立新的 layer：

a. 導覽至 source/infrastructure/lib/utils/lambda-aspects.ts 檔案中的 LambdaAspects類別。

b. 遵循如何擴展檔案中提供的 Lambda 觀點類別功能的指示 （例如新增 getOrCreateLangchainLayer方法）。若要使用此新方法 （例如 getOrCreateCustomLLMLayer)，請同時更新 source/infrastructure/lib/utils/constants.ts 檔案中的LLM_LIBRARY_LAYER_TYPES列舉。

3. 擴展 chat Lambda 函數以實作新供應商的建置器、用戶端和處理常式。

   source/lambda/chat 包含不同 LLMs 的 LangChain 連線，以及建置這些 LLMs的支援類別。這些支援類別遵循建置器和物件導向設計模式來建立 LLM。

   每個處理常式 （例如 bedrock_handler.py) 會先建立用戶端、檢查環境是否有所需的環境變數，然後呼叫 get_model方法來取得 LangChain LLM 類別。接著會呼叫 產生方法，以叫用 LLM 並取得其回應。LangChain 目前支援 Amazon Bedrock 的串流功能，但不支援 SageMaker AI。根據串流或非串流功能，呼叫適當的 WebSocket 處理常式 (WebsocketStreamingCallbackHandler 或 WebsocketHandler)，以使用 post_to_connection方法將回應傳回 WebSocket 連線。

   clients/builder 資料夾包含的類別可協助使用 Builder 模式建置 LLM Builder。首先，會從 DynamoDB 組態存放區use_case_config擷取 ，存放要建構之知識庫、對話記憶體和模型類型的詳細資訊。它還包含相關的模型詳細資訊，例如模型參數和提示。建置器接著有助於遵循建立知識庫、建立對話記憶體以維護 LLM 的對話內容、為串流和非串流案例設定適當的 LangChain 回呼，以及根據提供的模型組態建立 LLM 模型。當您從部署儀表板部署使用案例時 （或使用者在沒有部署儀表板的獨立使用案例堆疊部署中提供使用案例時），DynamoDB 組態會在建立使用案例時存放。

   clients/factories 子資料夾可協助根據 LLM 組態設定適當的對話記憶體和知識庫類別。這可讓您輕鬆地延伸到您希望實作支援的任何其他知識庫或記憶體類型。

   shared 子資料夾包含建置器在工廠內執行個體化的知識庫和對話記憶體的特定實作。它還包含在 LangChain 內呼叫的 Amazon Kendra 和 Amazon Bedrock 知識庫擷取器，以擷取 RAG 使用案例的文件，以及 LangChain LLM 模型使用的回呼。

   LangChain 實作使用 LangChain 表達式語言 (LCEL) 來組合對話鏈。RunnableWithMessageHistory類別用於維護具有自訂 LCEL 鏈的對話歷史記錄，啟用功能，例如傳回來源文件，以及使用傳送至知識庫的重述 （或取消歧義） 問題，以同時傳送至 LLM。

   若要建立自訂提供者的自有實作，您可以：

a. 複製 bedrock_handler.py 檔案並建立您的自訂處理常式（例如 custom_handler.py)，這會建立您的自訂用戶端（例如 CustomProviderClient)（在下列步驟中指定）。

b. bedrock_client.py 在用戶端資料夾中複製 。將其重新命名為 custom_provider_client.py（或您的特定模型提供者名稱，例如 CustomProvider)。適當地命名其中的類別，例如CustomProviderClient繼承 的 LLMChatClient。

   您可以使用 提供的方法，LLMChatClient或撰寫自己的實作來覆寫這些實作。

   get_model 方法會建置 CustomProviderBuilder（請參閱下列步驟），並使用建置器步驟呼叫建構聊天模型construct_chat_model的方法。此方法在建置器模式中充當 Director。

c. 將其複製clients/builders/bedrock_builder.py至 custom_provider_builder.py，並將其重新命名為繼承 LLMBuilder CustomProviderBuilder的 類別 (llm_builder.py)。您可以使用 LLMBuilder 提供的方法，或撰寫您自己的實作來覆寫這些實作。建置器步驟會在用戶端的 construct_chat_model方法內依序呼叫，例如 set_model_defaults、set_knowledge_base和 set_conversation_memory。

   該set_llm_model方法將使用之前稱為 的方法設定的所有值來建立實際 LLM 模型。具體而言，您可以根據從 DynamoDB 中的 LLM 組態擷取rag_enabled variable的 建立 RAG () CustomProviderRetrievalLLM或非 RAG (CustomProviderLLM) LLM。

   此組態會在 LLMChatClient類別的 retrieve_use_case_config方法中擷取。

d. 根據您是否需要 RAG 或非 RAG 使用案例，在llm_models子資料夾中實作您的 或 CustomProviderLLMCustomProviderRetrievalLLM實作。實作這些模型的大多數功能分別在非 RAG BaseLangChainModel和 RAG 使用案例的 和 RetrievalLLM類別中提供。

   您可以複製 llm_models/bedrock.py 檔案並進行必要的變更，以呼叫參考自訂提供者的 LangChain 模型。例如，Amazon Bedrock 使用ChatBedrock類別來建立使用 LangChain 的聊天模型。

   產生方法會使用 LangChain LCEL 鏈產生 LLM 回應。

   您也可以使用 get_clean_model_params方法來淨化每個 LangChain 或模型需求的模型參數。

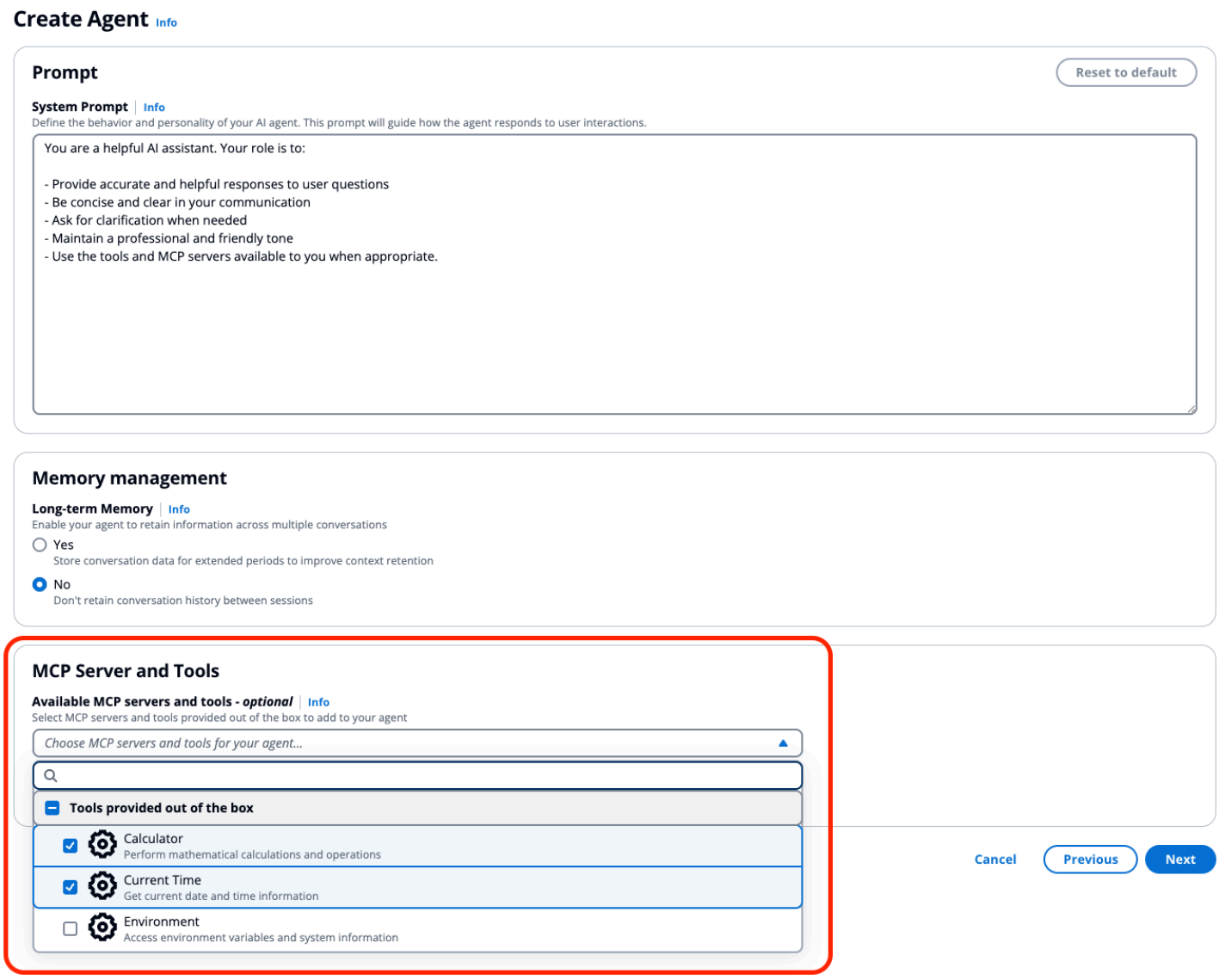
展開支援的 Strands 工具

解決方案可讓您建置和部署 MCP 伺服器、AI 代理器和多代理程式工作流程。在客服人員建置器體驗中，您可以連接 MCP 伺服器，為客服人員提供額外的功能。除了 MCP 伺服器之外，您還可以利用 Strands（解決方案所使用的基礎架構）提供的內建工具。

現成可用的解決方案已預先設定下列 Strands 工具：

- 目前時間（預設為啟用）
- 計算器（預設為啟用）
- Environment

客服人員建置器精靈中的 MCP 伺服器和工具選擇，顯示內建 Strands 工具



若要使用其他 Strands 工具擴展您的代理程式，請遵循本節中概述的四個步驟。

步驟 1：尋找 Strands 工具

瀏覽可用的 Strands 工具，以識別您要使用的工具。每個工具都有特定的功能和組態需求。

例如，若要新增 Amazon Bedrock 知識庫擷取功能，您可以使用擷取工具。

步驟 2：更新 SSM 參數

若要在 Agent Builder 部署 UI 中提供工具，請更新 AWS Systems Manager 參數存放區參數，以定義支援哪些 Strands 工具。

1. 導覽至您 AWS 帳戶中的 AWS Systems Manager 參數存放區。

2. 找到 參數： `/gaab/<stack-name>/strands-tools`
3. 使用下列 JSON 結構將您的工具組態新增至現有清單的結尾：

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

欄位	Description
name	客服人員建置器使用者介面中顯示的顯示名稱
description	工具功能的簡短描述
value	Strands 工具套件中定義的確切工具名稱
category	UI 中分組工具的組織類別
isDefault	根據預設，是否應為新客服人員啟用工具

步驟 3：設定環境變數

許多 Strands 工具需要環境變數才能進行組態。您可以透過兩種方式設定這些變數：

選項 1：AgentCore 執行期上的直接組態

使用所需的環境變數，直接在 Amazon Bedrock AgentCore 執行期更新部署的代理程式。

選項 2：部署精靈中的模型參數

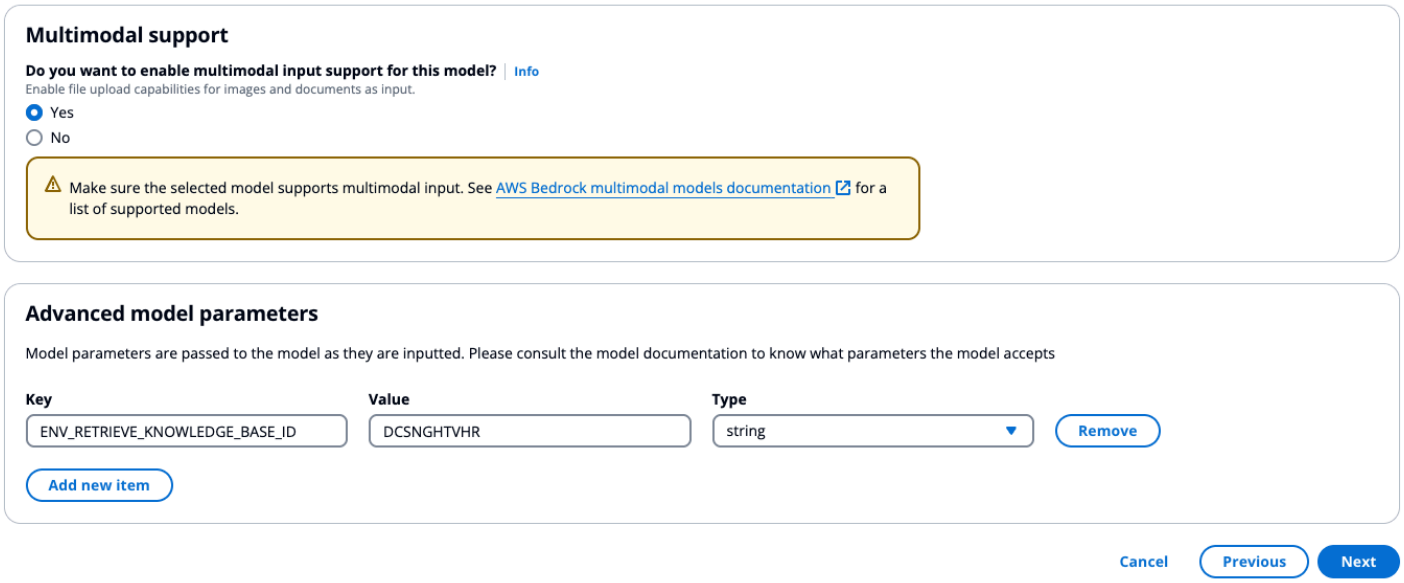
在客服人員建置器精靈的模型選取步驟期間，使用模型參數區段新增環境變數。遵循命名慣例的環境變數`ENV_<ALL_CAPS_TOOL_NAME>_<env_variable_name>`會在執行時間自動載入代理程式的執行環境，做為 `<env_variable_name>`。

例如：

- `ENV_RETRIEVE_KNOWLEDGE_BASE_ID` 成為 `KNOWLEDGE_BASE_ID`

- ENV_RETRIEVE_MIN_SCORE 成為 MIN_SCORE

顯示 ENV_RETRIEVE_KNOWLEDGE_BASE_ID 組態的進階模型參數區段



請參閱特定工具的文件或原始程式碼，以識別所需的環境變數。對於擷取工具，您可以在原始程式碼中找到組態選項。

步驟 4：新增 IAM 許可

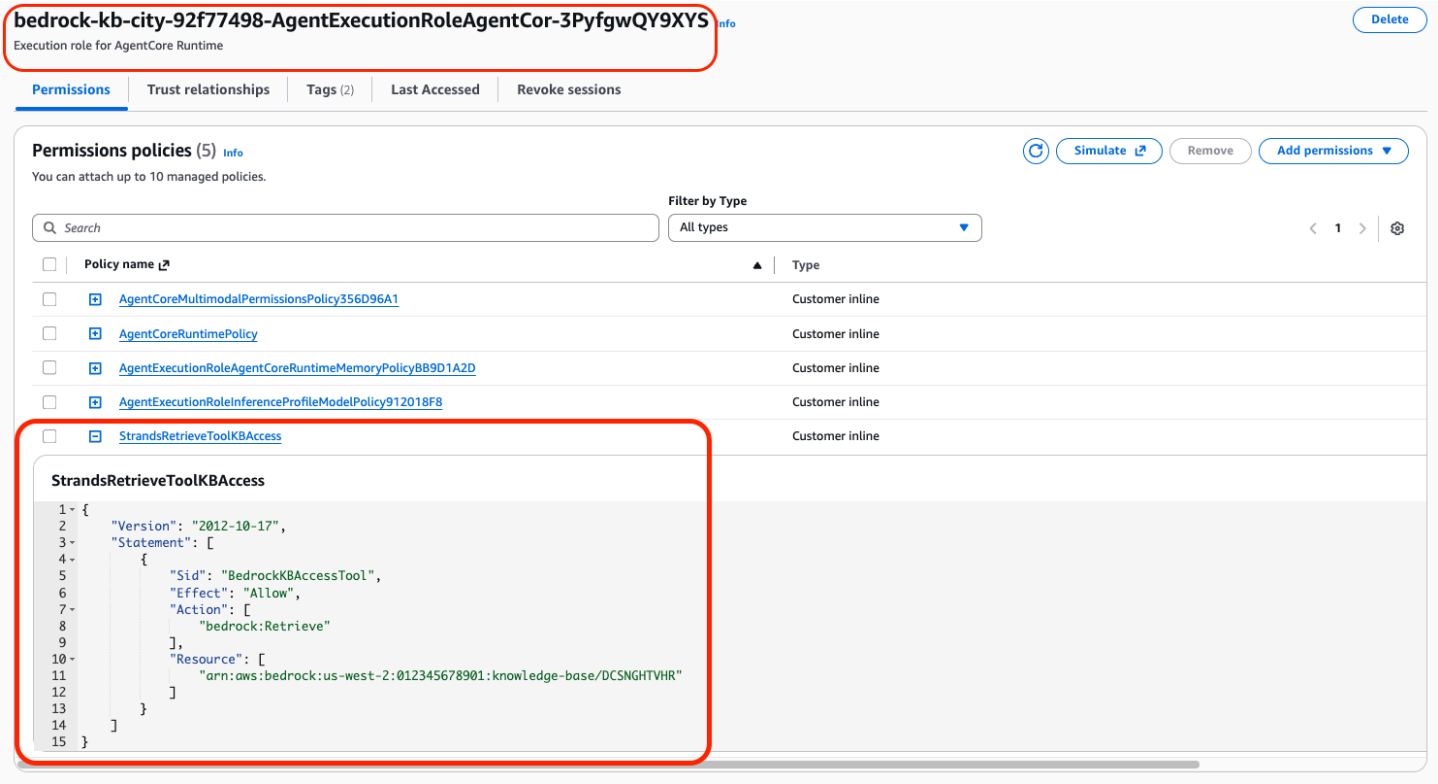
手動將任何必要的 IAM 許可新增至 AgentCore 執行期執行角色，以允許代理程式使用 工具。

例如，若要搭配 Amazon Bedrock 知識庫使用擷取工具：

1. 導覽至 AWS 帳戶中的 IAM 主控台。
2. 尋找代理程式的 AgentCore 執行期執行角色。
3. 新增下列許可：

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

IAM 主控台顯示連接到 AgentCore 執行期執行角色的 StrandsRetrieveToolKBAccess 政策

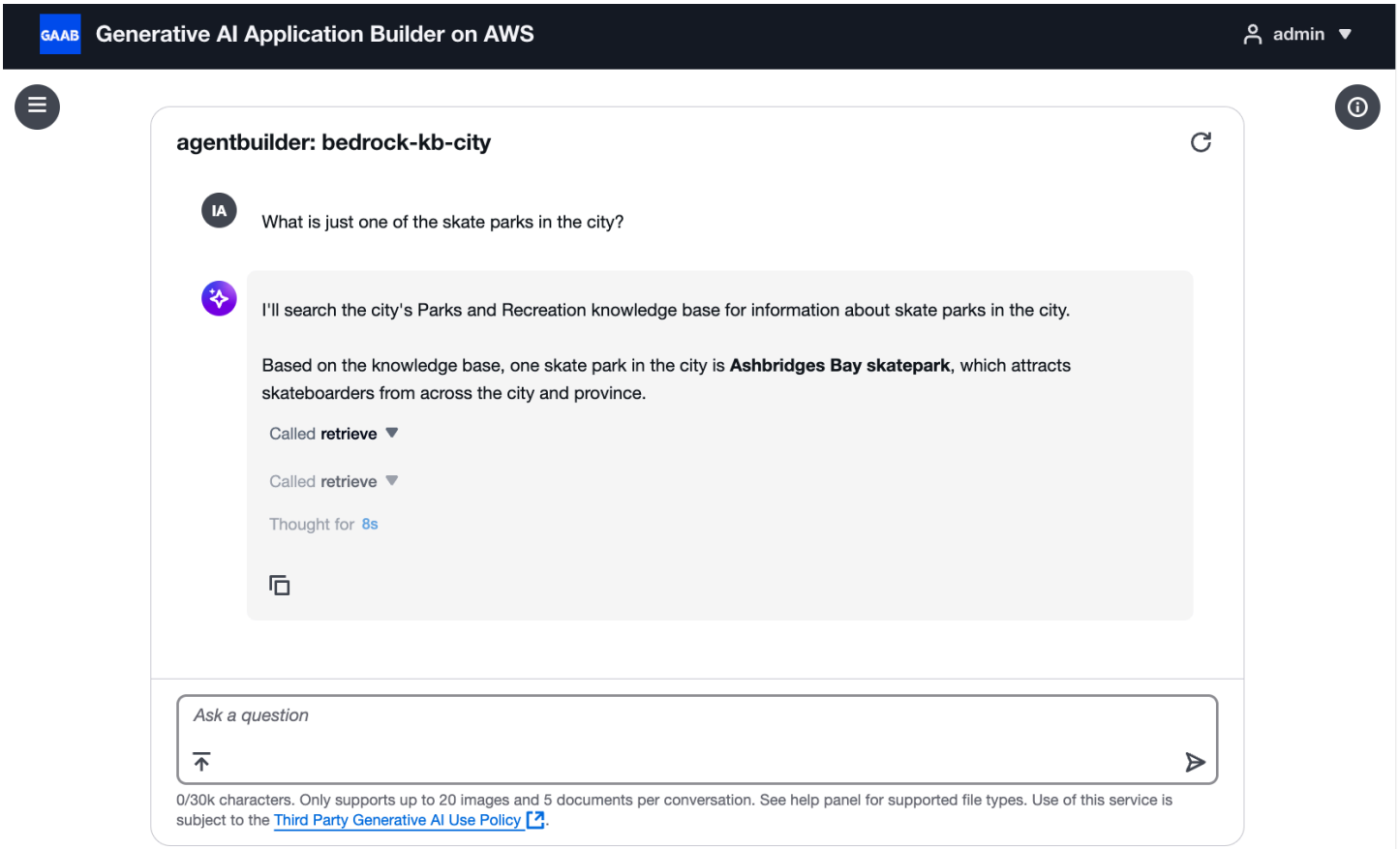


所需的特定許可會根據工具而有所不同。請參閱工具的文件和 AWS 服務文件，以判斷適當的 IAM 許可。

步驟 5：測試代理程式

完成組態步驟後，測試您的代理程式以確認工具是否正常運作。您應該會在客服人員的執行日誌和回應中看到工具叫用。

客服人員成功使用擷取工具來回答有關滑板公園的問題



> **Note**
>
> 如需可用 Strands 工具及其功能的完整清單，請參閱 Strands 社群工具文件。

擴展支援的知識庫和對話記憶體類型

若要新增對話記憶體或知識庫的實作，請在 `shared` 資料夾中新增所需的實作，然後編輯工廠和適當的列舉，以建立這些類別的執行個體。

當您提供存放在參數存放區中的 LLM 組態時，將會為您的 LLM 建立適當的對話記憶體和知識庫。例如，`ConversationMemoryType`當 指定為 DynamoDB 時，會建立 執行個體 `DynamoDBChatMessageHistory`（可在 內取得`shared_components/memory/ddb_enhanced_message_history.py`）。當 `KnowledgeBaseType` 指定為 Amazon Kendra 時，會建立 執行個體 `KendraKnowledgeBase`（可在 內取得`shared_components/knowledge/kendra_knowledge_base.py`）。

建置和部署程式碼變更

使用 `npm run build`命令建置程式。解決任何錯誤後，請執行 `cdk synth`以產生範本檔案和所有 Lambda 資產。

1. 您可以使用`0/stage-assets.sh`指令碼，將任何產生的資產手動暫存到帳戶中的預備儲存貯體。
2. 使用下列命令來部署或更新平台：

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

任何其他 AWS CloudFormation 參數也應該與 AdminUserEmail 參數一起提供。

自訂指南

管理 Cognito 使用者集區

部署儀表板時，會建立 Amazon Cognito 使用者集區以及管理員使用者，以提供應用程式的身分驗證。此使用者集區會跨部署儀表板和所有使用案例共用。部署儀表板時建立的管理員使用者會自動獲得使用儀表板部署之所有使用案例的存取權。此機制是透過 Amazon Cognito 使用者集區群組提供。

從儀表板部署使用案例時，如果提供電子郵件，則會在共用使用者集區中建立使用者，以及針對特定使用案例名為 的使用者群組。然後，新建立的使用者會新增至 群組，授予使用者存取使用案例的權限。

如果您想要將其他使用者新增至指定的使用案例，您可以透過在 Cognito 使用者集區中建立使用者，並將他們新增至與您希望使用者存取的使用案例（些）對應的群組來達成此目標。如需step-by-step指南，請參閱 AWS 管理主控台中的建立新使用者。

同樣地，如果您想要建立其他管理員使用者，則必須建立新的使用者，並將其新增至使用者集區中的管理員群組。

使用者名稱是透過在 之前取得所提供電子郵件的一部分來建立@，並附加產生的使用案例 UUID（或在管理員使用者`-admin`的情況下）。

在群組索引標籤中，您可以看到已使用使用案例名稱（如精靈中提供的）和使用案例 UUID 自動建立每個使用案例的管理員群組和群組。

API 參考

本節提供解決方案的 API 參考。

部署儀表板

REST API	HTTP 方法	功能	授權來電者
/deployments	GET	取得所有部署。	Amazon Cognito 驗證的 JWT 字符
/deployments	POST	建立新的使用案例部署。	Amazon Cognito 驗證的 JWT 字符
/deployments/{useCaseId}	GET	取得單一部署的部署詳細資訊。	Amazon Cognito 驗證的 JWT 字符
/deployments/{useCaseId}	PATCH	更新指定的部署。	Amazon Cognito 驗證的 JWT 字符
/deployments/{useCaseId}	DELETE	刪除指定的部署。	Amazon Cognito 驗證的 JWT 字符
/model-info/use-case-types	GET	取得部署可用的使用案例類型	Amazon Cognito 驗證的 JWT 字符
/model-info/{useCaseType}/providers	GET	取得指定使用案例類型的可用模型提供者	Amazon Cognito 驗證的 JWT 字符
/model-info/{useCaseType}/{providerName}	GET	取得特定提供者和使用案例類型可用的模型 IDs	Amazon Cognito 驗證的 JWT 字符
/model-info/{useCaseType}/{providerName}/{modelId}	GET	取得指定模型的相關資訊，包括預設參數。	Amazon Cognito 驗證的 JWT 字符

> **Note**
>
> OpenAPI 和 Swagger 檔案也可以從 API Gateway 匯出，以便與 API 整合。請參閱從 API Gateway 匯出 REST API。

POST 和 PATCH 承載

如需端點的 POST 承載範例/deployments，請參閱下文，這會建立新的使用案例。

```
{
 "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
 }
 }

}
```

對於更新，結構與上述相同，但有一些注意事項：

- 無法變更使用案例名稱
- 使用案例只有在部署在 VPC 之後，才能變更安全群組和子網路。VPC 本身無法變更。
- 如果已為您建立 Kendra 索引做為知識庫，則無法變更該索引的組態 （ 例如 KendraIndexName、QueryCapacityUnits)

共用使用案例 APIs

下列 REST API 端點適用於文字和 Bedrock 代理程式使用案例：

REST API	HTTP 方法	功能	授權來電者
`/details/{useCaseConfigKey}`	`GET`	取得特定使用案例的組態詳細資訊。	Amazon Cognito 驗證的 JWT 字符

WebSocket API	功能	授權來電者
`/$connect`	啟動 WebSocket 連線並驗證使用者。	Amazon Cognito 驗證的 JWT 字符
`/$disconnect`	當 WebSocket 連線中斷時呼叫的端點。	Amazon Cognito 驗證的 JWT 字符

使用案例詳細資訊 API

API 端點擷取特定使用案例相關資訊的詳細資訊：

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

此端點會傳回特定使用案例的組態詳細資訊，包括模型參數、知識庫設定和其他部署資訊。它需要 Amazon Cognito 驗證的 JWT 權杖才能進行授權。

文字使用案例

WebSocket API	功能	授權來電者
`/sendMessage`	將使用者的聊天訊息傳送至 WebSocket，以使用設定的 LLM 體驗進行處理。	Amazon Cognito 驗證的 JWT 字符

REST API	HTTP 方法	功能	授權來電者
/feedback/{useCaseId}	POST	提交特定使用案例的使用者意見回饋。	Amazon Cognito 驗證的 JWT 字符

sendMessage 承載

如果您直接與 /sendMessage API 整合，則必須遵守下列請求和回應承載格式。

請求承載

```
{
 "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
}
```

參數名稱	Type	說明
動作	String	目前我們僅支援 WebSocket 上的「sendMessage」動作
問題	String	要傳送至 LLM 的使用者輸入
conversationId	String	識別對話的 UUID。如果未提供，則會建立新的對話，並在回應中傳回 conversationId。該對話中的所有後續訊息（您希望保留歷史記錄/內容的位置）應在該對話中提供 conversationId。
promptTemplate	String 【選用】	覆寫此訊息的提示範本。如果為空或未提供，將預設為部署

參數名稱	Type	說明
		時設定的提示。必須為指定的組態指定適當的預留位置（即非 RAG Sagemaker AI 部署的 {history} 和 {input}，如果所有部署都使用 RAG，則新增 {context}。
authToken	`String` 【選用】	從 cognito 驗證流程取得的 accessToken。使用角色型存取控制 (RBAC) 調用針對 RAG 設定的聊天 Websocket 端點時，這是必要的。此 JWT 字符中的 cognito：groups 宣告清單用於控制對 Kendra 索引中文件的存取。非 RAG 使用案例不需要此參數。停用 RBAC 的 RAG 使用案例也不需要。

回應承載

問題回應

WebSocket API 將以 1（如果停用串流）或多個（如果啟用串流）JSON 物件回應，每個查詢的結構如下。

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

參數名稱	Type	說明
資料	`String`	如果啟用串流，則為 LLM 的回應區塊，或整個回應。如果使用串流，則會傳送此格式且資料內容為 END_CONVE

參數名稱	Type	說明
		RSATION 的回應，以指出單一問題的回應結束。
conversationId	String	此 sourceDocument 回應所屬的對話 ID。

來源文件回應

如果您已將 RAG 使用案例設定為傳回來源文件，您也會在用於建立回應的每個來源文件的每個回應結束時收到下列承載。

```
{
 "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"
}
```

參數名稱	Type	說明
摘錄	String	來源文件的摘錄。
位置	String	來源文件的位置。這取決於使用的資料來源和知識庫類型，但可能是 s3 URIs 或網站之類的內容。
分數	Number | String	文件對應到所問問題的可信度。對於 Bedrock，這將是從 0 到 1 的浮點數，對於 Kendra 則為字串（例如 HIGH、LOW 等）。

參數名稱	Type	說明
document_title	String	傳回來源文件的標題。僅適用於使用 Kendra 時。
document_id	String	傳回來源文件的 ID。僅適用於使用 Kendra 時。
additional_attributes	String	此欄位將包含文件上的所有其他屬性，如擷取時的知識庫所自訂。
conversationId	String	此 sourceDocument 回應所屬的對話 ID。

意見回饋 API 承載

以下是端點的 POST 承載範例`/feedback/{useCaseId}`，它會針對特定使用案例提交使用者意見回饋：

```
{
  "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"
  ]
}
```

Bedrock Agent 使用案例

WebSocket API	功能	授權來電者
`/invokeAgent`	將使用者的訊息傳送至 WebSocket，以便使用設定的代理程式進行處理。	Amazon Cognito 驗證的 JWT 字符

invokeAgent 承載

如果您直接與 整合`/invokeAgent` API，則必須遵守下列請求和回應承載格式。

請求承載

```
{
"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
}
```

參數名稱	Type	說明
動作	`String`	我們僅支援 WebSocket 上的 `invokeAgent` 動作。
inputText	`String`	要傳送至 LLM 的使用者輸入。
conversationId	`String[Optional]`	唯一識別對話的 UUID。如果您不提供此值，解決方案會建立新的對話，並在回應中傳回 conversationId。該對話中的所有後續訊息（您要保留歷史記

參數名稱	Type	說明
		錄和內容的位置） 都會在該處提供 conversationId。
authToken	String[Optional]	從 Amazon Cognito 驗證流程取得的 accessToken。不需要此參數。如果您提供它，則會驗證 JWT 字符。這有助於更輕鬆地擴展此解決方案。

回應承載

問題回應

WebSocket API 將以一個 （如果停用串流） 或多個 （如果啟用串流） JSON 物件回應，每個查詢的結構如下。

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

參數名稱	Type	說明
資料	String	來自客服人員調用的回應。
conversationId	String	對話的 ID。

參考資料

本節包含此解決方案的資料收集資訊、相關資源的指標，以及有助於此解決方案的建置器清單。

支援的 LLM 供應商

解決方案可與下列 LLM 提供者整合：

1. Amazon Bedrock
   - 文件：https：//https://aws.amazon.com/bedrock/
   - 支援的模型：
     - Amazon
       - Nova Lite
       - Nova Micro
       - Nova Pro
     - AI21 Labs
       - Jamba 1.5 Mini
       - Jamba 1.5 Large
     - Anthropic
       - Claude v3 Haiku
       - Claude v3.5 Sonnet
       - Claude v3.7 Sonnet（透過使用推論設定檔）
     - Cohere
       - Command R
       - Command R+
     - Deepseek
       - Deepseek-R1（透過使用推論設定檔）
     - Meta
       - Llama 3
       - Llama 3.2（透過使用推論設定檔）
     - Mistral AI
       - Mistral 7B Instruct

    - Mistral 8x7B 指示
  - 跨區域推論
    - 能夠使用與部署儀表板在相同區域中定義的推論設定檔
2. Amazon SageMaker AI
   - 文件：https：//https://aws.amazon.com/sagemaker/
   - 支援的模型：文字轉文字模型

如需最新的模型參數、最佳實務和建議用途，請參閱模型提供者的文件。

資料收集

此解決方案會將使用此解決方案的操作指標傳送給 AWS (「資料」)。我們使用此資料來更好地了解客戶如何使用此解決方案和相關的服務和產品。AWS 收集此資料受 AWS 隱私權聲明約束。

貢獻者

- Tarek Abdunabi
- Majd Arbash
- George Bearden
- Mukit Bin 最小值
- Michael Connor
- Johny Duval
- Nihit Kasabwala
- Ahern Knox
- Simon Krol
- Michael Lin
- Tim Mekari
- Ibrahim Mohamed
- Omar Radwan Mohsen
- 詹姆士尼克森
- Dekshitha Ravikumar
- Jae Shim

- Ajay Swamy
- 清真塔哈
- Reet Takkar
- Dimitri Tchikatilov
- Jason Wreath
- Kamyar Ziabari

修訂

發佈日期：2023 年 10 月（上次更新日期：2025 年 1 月）

檢查 GitHub 儲存庫中的 CHANGELOG.md 檔案，以查看軟體的所有顯著變更和更新。變更日誌提供每個版本的改善和修正的清楚記錄。

注意

客戶有責任對本文件中的資訊進行自己的獨立評定。本文件：(a) 僅供參考，(b) 代表 AWS 目前的產品產品和實務，如有變更，恕不另行通知，且 (c) 不會從 AWS 及其附屬公司、供應商或授權方建立任何承諾或保證。AWS 產品或服務會以「原樣」提供，不做任何明示或暗示的保證、陳述或條件。AWS 對其客戶的責任和義務由 AWS 協議控制，本文件並非 AWS 與其客戶之間任何協議的一部分，也不會加以修改。

AWS 上的生成式 AI 應用程式建置器是根據 Apache 授權 2.0 版進行授權。

Important

AWS 上的生成式 AI 應用程式建置器可讓您透過使用您選擇的生成式 AI 模型，在 AWS 上建置和部署生成式人工智慧應用程式，包括您可以選擇使用該 AWS 不擁有或擁有任何控制權的第三方生成式 AI 模型 (「第三方生成式 AI 模型」)。

當您取得第三方生成式 AI 模型的使用授權 （例如，其服務條款、授權合約、可接受的使用政策和隱私權政策） 時，您對第三方生成式 AI 模型的使用受第三方生成式 AI 模型提供者提供給您的條款約束。

您有責任確保您使用第三方生成式 AI 模型時遵守管理它們的條款，以及適用於您的任何法律、規則、法規、政策或標準。

您也必須負責獨立評估您使用的第三方生成式 AI 模型，包括其輸出，以及第三方生成式 AI 模型提供者如何使用根據您的部署可能傳輸給他們的任何資料。根據您與 AWS 的協議，AWS 不對第三方生成式 AI 模型提供任何聲明、保證或保證，即「第三方內容」。根據您與 AWS 的協議，AWS 上的生成式 AI 應用程式建置器會以「AWS 內容」的形式提供給您。

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。