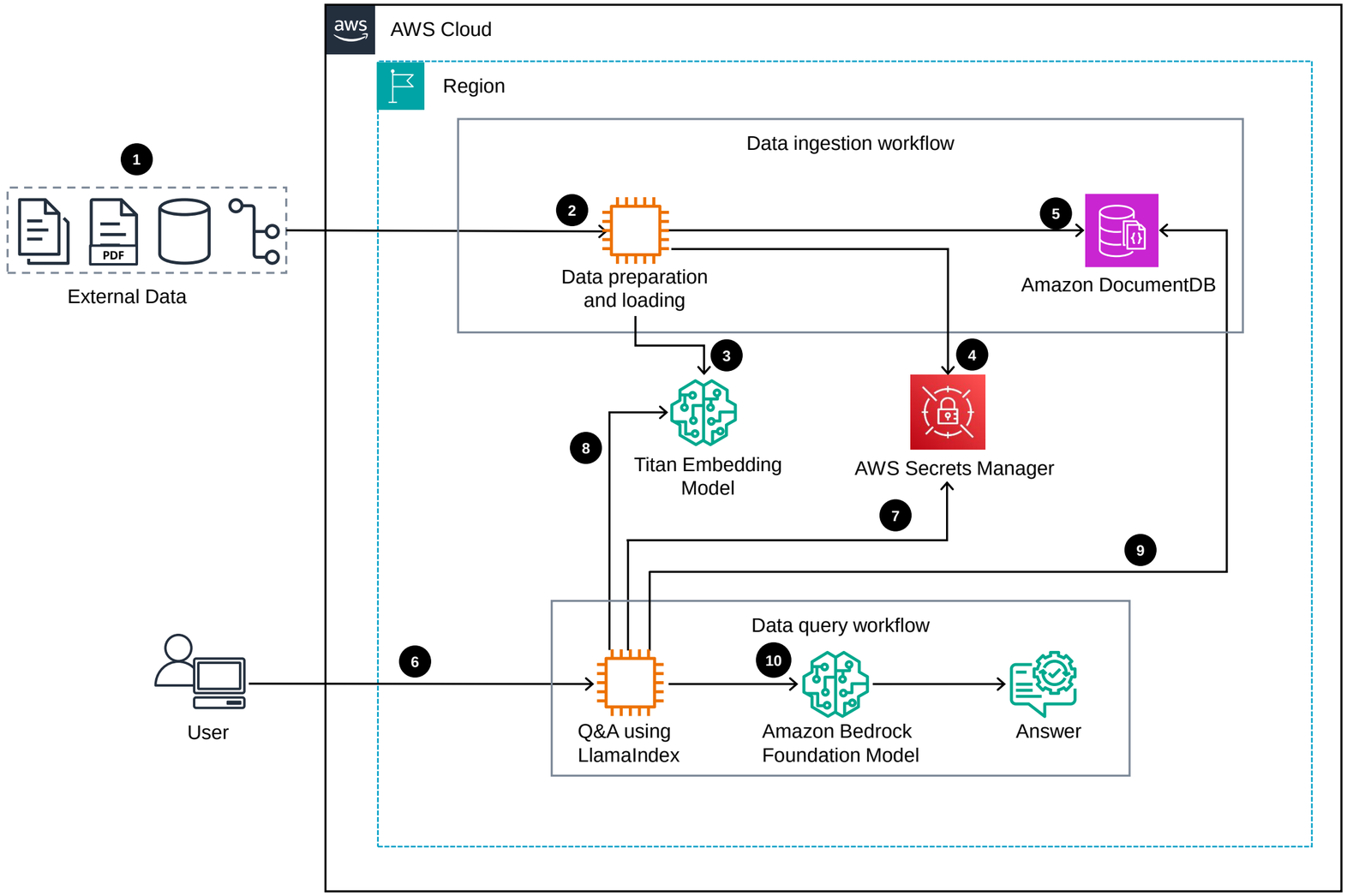
Guidance for Similarity Search-Based Retrieval Augmented Generation (RAG) on AWS

This architecture diagram illustrates how to process user queries and generate accurate, contextually relevant responses. It enhances a foundation model (FM) on Amazon Bedrock using Retrieval Augmented Generation (RAG); the vector search capabilities of Amazon DocumentDB and LlamaIndex enable more accurate and informed answers from a customized knowledge base.



1. The user uploads enterprise or external data, which lies outside of the large language model's (LLM) training data, to augment the trained model. It can come from various sources including APIs, databases, or document repositories.
2. The application hosted on **Amazon Elastic Compute Cloud** (Amazon EC2) preprocesses data by removing inconsistencies and errors, splitting large documents into manageable sections, and chunking the text into smaller, coherent pieces for easier processing.
3. The application generates text embeddings for relevant data using the Titan text embedding models on **Amazon Bedrock**.
4. The application fetches credentials from **AWS Secrets Manager** to connect to **Amazon DocumentDB (with MongoDB compatibility).**
5. The application creates a vector search index in **Amazon DocumentDB** and uses LlamaIndex to load the generated text embeddings along with other relevant information into an **Amazon DocumentDB** collection.
6. The user submits a natural language query for finding relevant answers to a web application.
7. The application fetches credentials from **Secrets Manager** to connect to **Amazon DocumentDB**.
8. The user's question is transformed into a vector embedding in the application using the same embedding model that was used during the data ingestion workflow.
9. The application passes the query to the LlamaIndex query engine. LlamaIndex is a data orchestration tool that helps with data indexing and querying. LlamaIndex performs a similarity search in the **Amazon DocumentDB** collection using the query embedding. The search retrieves the most relevant documents based on their proximity to the query vector.
10. The LlamaIndex query engine augments this retrieved information, along with the user's question, as a prompt to the LLM model on **Amazon Bedrock** to generate more accurate and informed responses.

Reviewed for technical accuracy December 12, 2024

AWS Reference Architecture