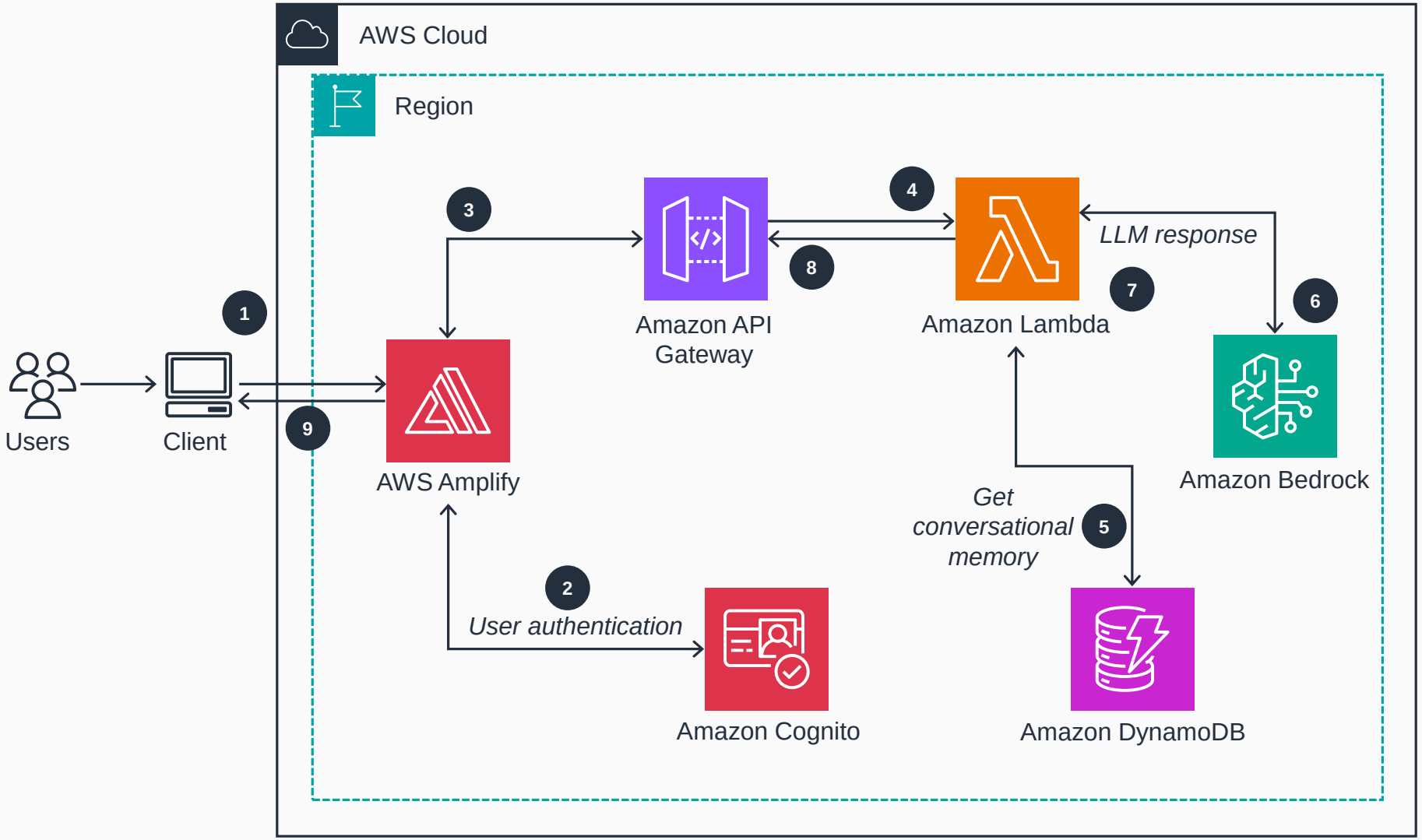
Guidance for Generative AI Assistant on AWS

This architecture diagram shows how to build a generative AI-powered application on AWS that summarizes web content and enables conversational interactions with that content.



1. Users access a web application based on React that is served by **AWS Amplify**. User inputs can be a query with a web link or a PDF document to summarize.
2. **Amazon Cognito** user pool authenticates users.
3. When a user inputs a message, the application sends a POST request to the **Amazon API Gateway REST** API.
4. **API Gateway** then routes the message to the **AWS Lambda** function.
5. User conversations are stored in **Amazon DynamoDB**. Users can create separate threads for discussing separate topics in the **Amplify** web application.
6. The user input and conversation history is sent to **Amazon Bedrock** for large language model (LLM) response generation. Users can choose between **Amazon Nova Micro** or **Anthropic Claude Sonnet** foundation models.
7. When the response comes back from **Amazon Bedrock,** the **Lambda** function stores it in a **DynamoDB** table as conversational memory.
8. The **Lambda** function then sends back a response to the user with the same channel through **API Gateway**.
9. The **Amplify** frontend web application shows the responses.

Reviewed for technical accuracy May 14, 2025

AWS Reference Architecture