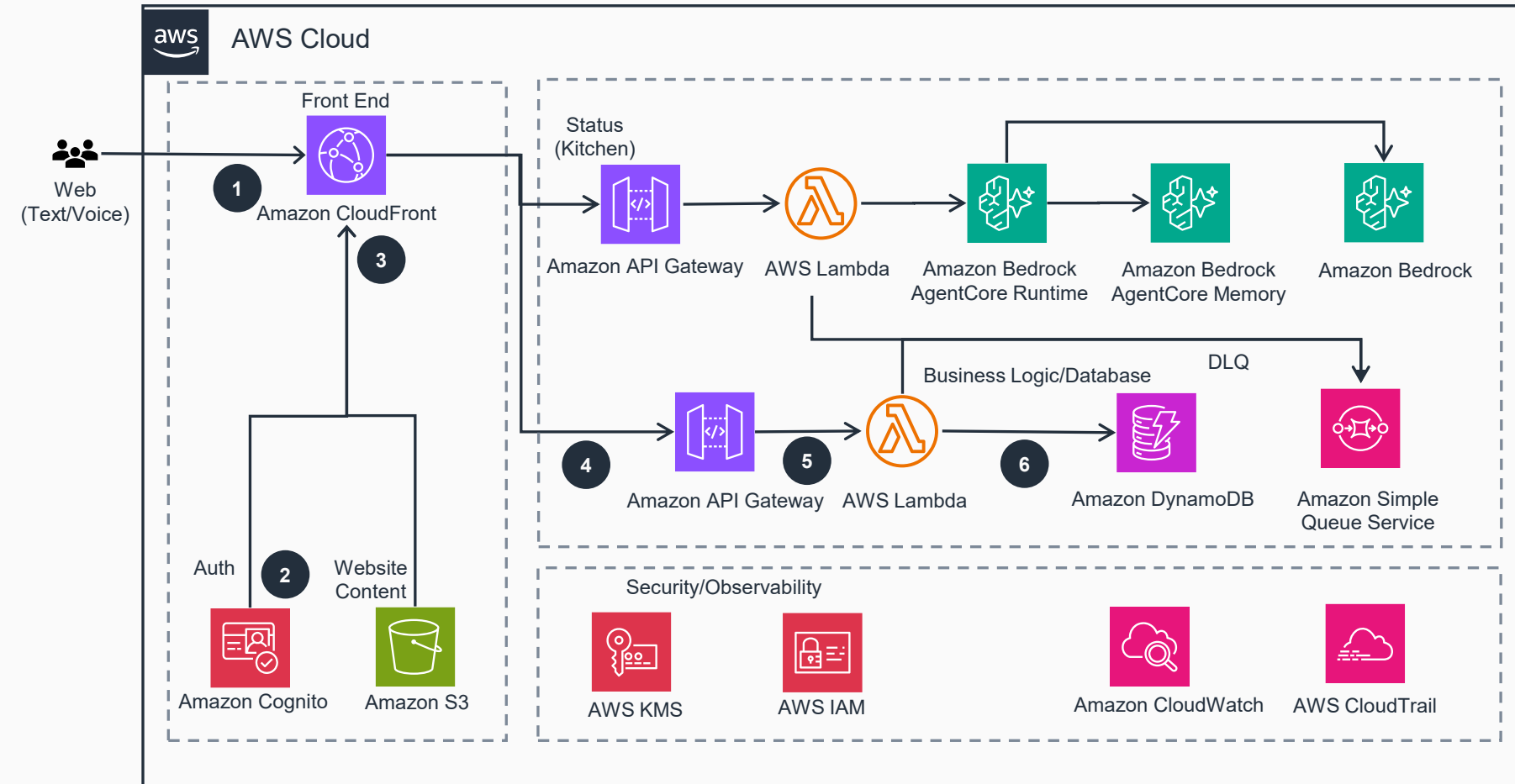


Guidance for Restaurant Visibility using Agents on AWS

This architecture diagram illustrates how to effectively support Restaurant Monitoring System with Gen AI voice/chat on AWS. It shows the key components and their interactions, providing an overview of the architecture's structure and functionality.

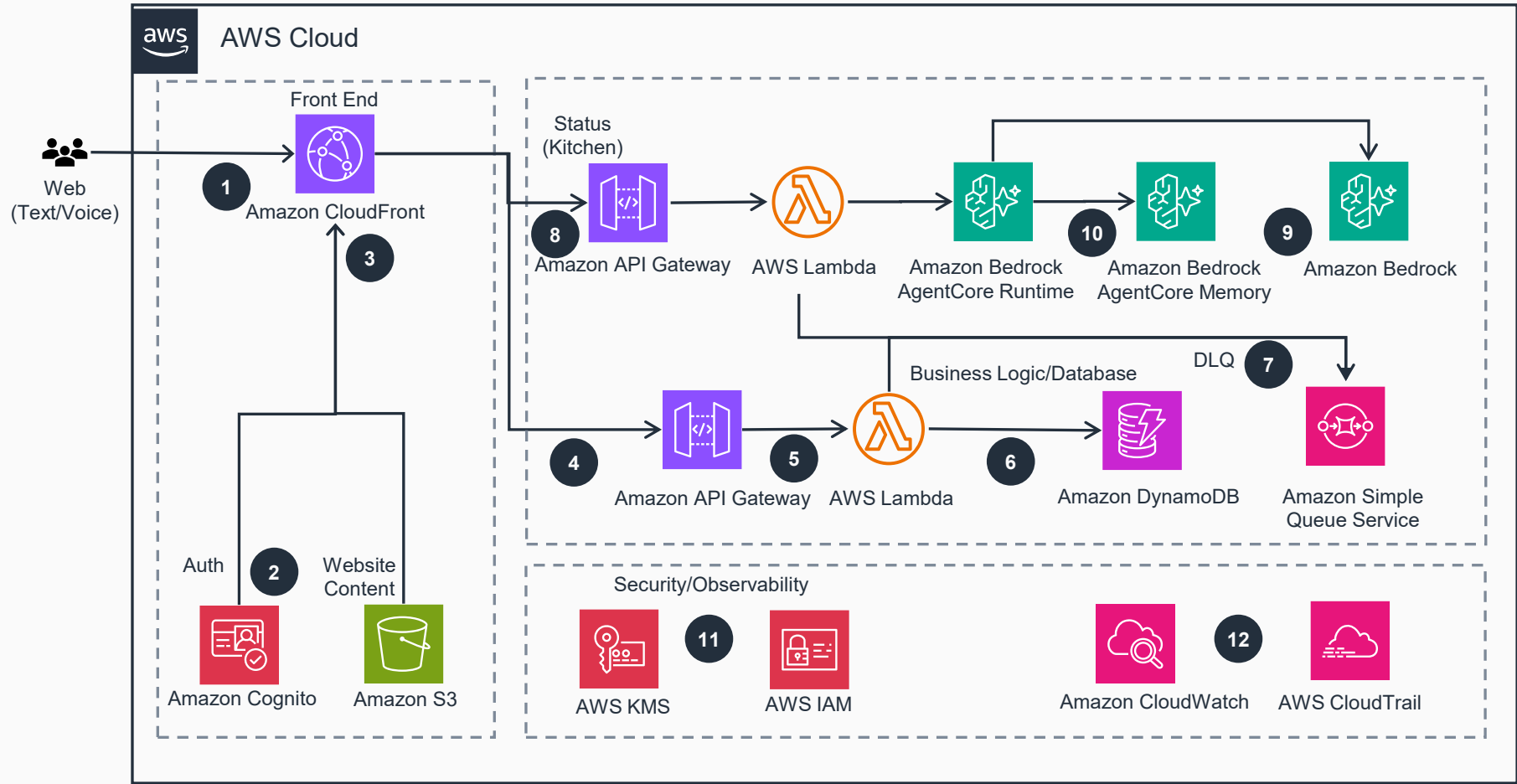


- 1 Customer invokes via text/voice on the web application deployed on the edge in **Amazon CloudFront**.
- 2 **Amazon Cognito** provides authentication and authorization for secure system access for **Amazon CloudFront** and **Amazon API Gateway**.
- 3 Customer requests to **Amazon CloudFront** are supported using website content stored in **Amazon Simple Storage Service**.
- 4 Customer requests from the web are routed as REST APIs to **Amazon API Gateway**.
- 5 **AWS Lambda** processes the **Amazon API Gateway** requests by reading appliance status and ticket status from **Amazon DynamoDB**, then passes the results to the agent orchestration layer.
- 6 **Amazon DynamoDB** functions as the central NoSQL database managing kitchen appliance and ticket information.



Guidance for Restaurant Visibility using Agents on AWS

This architecture diagram illustrates how to effectively support Restaurant Monitoring System with Gen AI voice/chat on AWS. It shows the key components and their interactions, providing an overview of the architecture's structure and functionality.



- 7 Amazon Simple Queue Service (Amazon SQS)** captures messages from failed AWS Lambda invocations in a Dead Letter Queue, enabling error isolation and reprocessing of unhandled requests.
- 8 Amazon API Gateway and AWS Lambda** support requests through the **Strands Agents SDK** deployed on **Amazon Bedrock AgentCore Runtime**, which provides a low-latency serverless environment with session isolation to execute the monitoring agent at scale.
- 9 Amazon Bedrock** provides access to foundation models as a fully managed service with built-in security and responsible AI controls.
- 10 Amazon Bedrock AgentCore Memory** manages session and long-term memory, enabling the agent to retain context across interactions such as ongoing appliance issues and conversation history.
- 11 AWS Key Management Service (AWS KMS)** supports encryption at rest for **Amazon DynamoDB**, **Amazon S3**, and **AWS Lambda environment variables**. **AWS Identity and Access Management (AWS IAM)** controls secure access across **AWS services with least-privilege policies**.
- 12 Amazon CloudWatch and AWS CloudTrail** provide system monitoring and audit logging across all services, including **Amazon Bedrock AgentCore** invocations and **Amazon DynamoDB** data access.