# **API Gateway Integration for Serverless Cloud Dictionary**

## **Project Overview**

I configured Amazon API Gateway to expose a REST API for the Lambda function, enabling the React frontend to dynamically fetch cloud term definitions. This integration completes the serverless architecture, providing a secure, scalable, and efficient backend interface for the Cloud Dictionary Application.

## **Implementation and Workflow**

1. **Create REST API**

* Created a REST API named CloudDictionaryAPIGateway in API Gateway.
* Configured the /get-definition resource to handle fetching definitions of cloud terms.

A computer screen with a white screen

AI-generated content may be incorrect.

Figure 1: API Gateway Console showing the created REST API CloudDictionaryAPIGateway.

1. **Lambda Proxy Integration**

* Connected the /get-definition resource to the FetchTermFromDynamoDB Lambda function using Lambda Proxy Integration, allowing query parameters, headers, and request body to pass directly to Lambda.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2: Configuring /get-definition resource with Lambda Proxy Integration in API Gateway.

1. **Enable CORS**

* Enabled Cross-Origin Resource Sharing (CORS) to allow secure communication between the React frontend and the API.

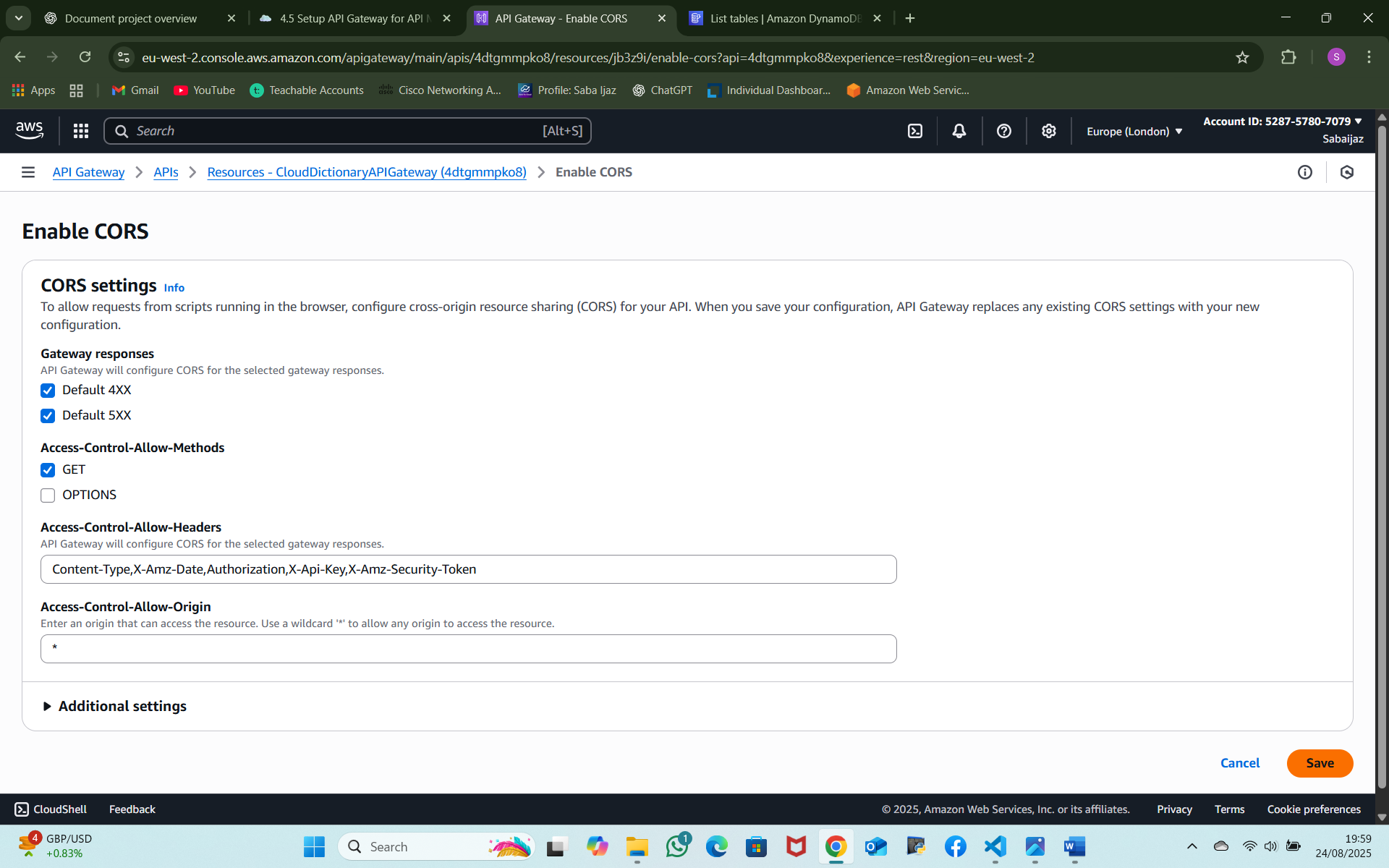


Figure 3: Enabling CORS on the/get-definition resource for frontend access.

1. **Deploy API**

* Deployed the API to a stage named dev and obtained the Invoke URL for integration with the frontend.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 4: Deployment stage dev with the API Invoke URL visible.

1. **Frontend Integration and Testing**

* Updated the App.js file in the React application with the API Gateway URL.
* Pushed changes to GitHub; AWS Amplify automatically triggered a new deployment.
* Tested the application to confirm that term searches return correct definitions from DynamoDB.

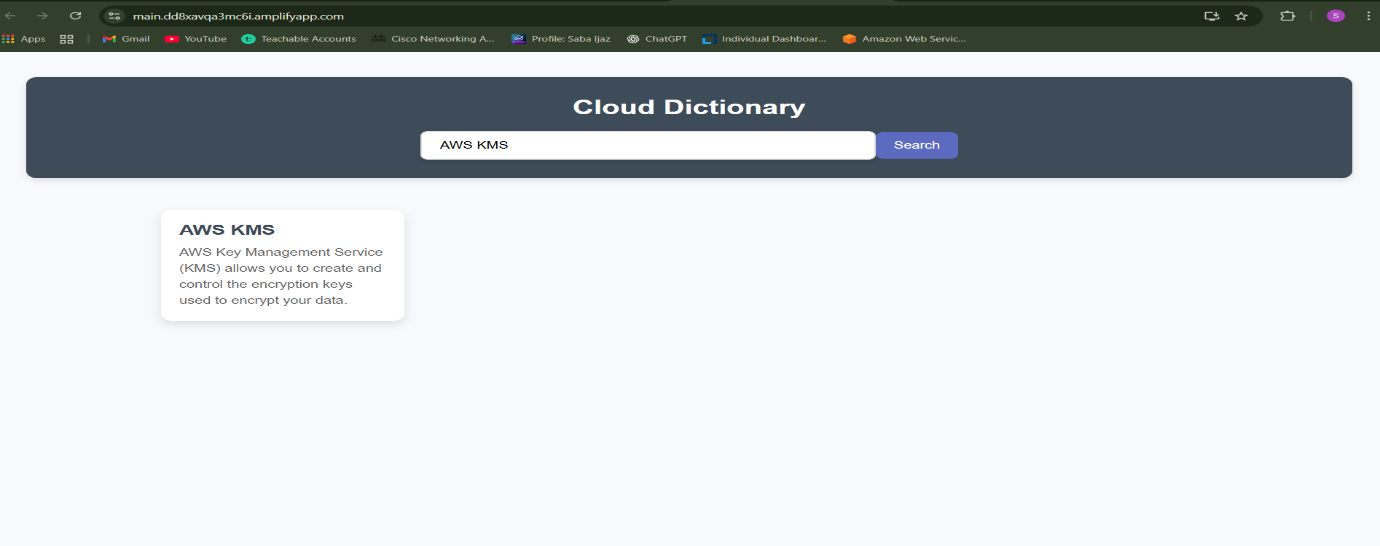


Figure 5: React application testing the API by searching for “AWS KMS” and receiving the correct definition.

**Outcomes**

The API Gateway now provides a serverless, scalable interface for the Lambda backend. Users can retrieve cloud term definitions in real time, and the React frontend communicates seamlessly with AWS services. This demonstrates end-to-end integration of serverless components for a functional Cloud Dictionary Application.