

Multilingual Voice Assistant using AWS Transcribe & AWS Translate

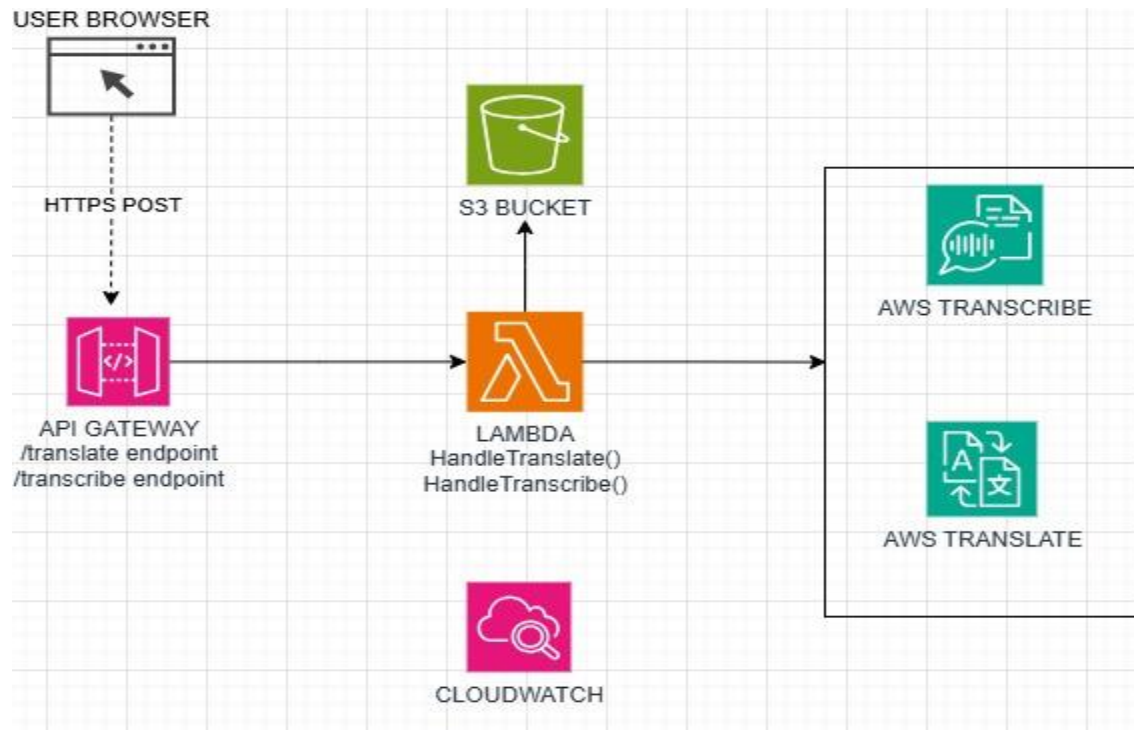
Project Index

1. Project Overview
2. Architecture Diagram
3. AWS Services Used
4. Frontend Workflow
5. Backend Workflow
6. Deployment With Terraform
7. Cost Considerations
8. Future Enhancements
9. Proof of Concept

1. Project Overview

This project implements a multilingual voice assistant that allows users to record audio in the browser, a base64-encoded audio is sent through an API Gateway endpoint to a Lambda function, and have the audio stored in S3 and transcribed using Amazon Transcribe. The Lambda function polls the transcription job until completion, retrieves the transcription JSON from an S3 URI using Node.js HTTPS module, extracts the text, and returns it to the frontend. The user can then submit the transcribed text to another API endpoint where AWS Translate processes it and returns the translated output. This system provides a streamlined, serverless workflow for audio recording, speech-to-text processing, and multilingual translation.

2. Architecture Diagram



Transcription Flow

Microphone → MediaRecorder API → Base64 Encoding → API Gateway (/transcribe) → Lambda (handleTranscribe) → S3 Upload → AWS Transcribe Job → Polling Loop → HTTPS Fetch Transcript → Parse JSON → Lambda → API Gateway → Browser

Translation Flow

Translate button → API Gateway (/translate) → Lambda (handleTranslate) → AWS Translate → Lambda → API Gateway → Browser

3. AWS Services Used

- **API Gateway** – REST endpoints
- **Lambda** – Node.js functions for transcription & translation
- **S3** – audio storage and transcription job output
- **Amazon Transcribe** – speech-to-text

- **Amazon Translate** – language translation
- **Cloudwatch** - For logging to troubleshoot issues

4. Frontend Workflow

Record audio

- **Use MediaRecorder API**
- **Convert audio Blob to base64**

Transcribe UI

- Button: **Record and stop to transcribe**
- Text area: **Shows transcription**

Translate

- Button: **Translate**
- **Shows translated output**

5. Backend Workflow

For /transcribe

1. API Gateway receives base64 audio and is sent to the lambda function.
2. Lambda decodes base64 audio to a file.
3. Uploads to S3.
4. Starts Transcribe job.
5. Polls transcription job till completion.
6. When completed, job output contains an S3 URI.
7. Lambda uses **https** module to fetch JSON output from the URI.
8. Extracts results (transcribed data).
9. Returns transcription back to the user interface.

For /translate

1. Takes text input from frontend.
2. Calls Amazon Translate.
3. Returns translated text.

6. Deployment With Terraform

The entire serverless architecture, including the S3 bucket, Lambda functions, API Gateway endpoints, IAM roles, and necessary permissions can be provisioned and managed using Terraform. Full configuration files can be found in this [GitHub repo](#).

7. Cost Considerations

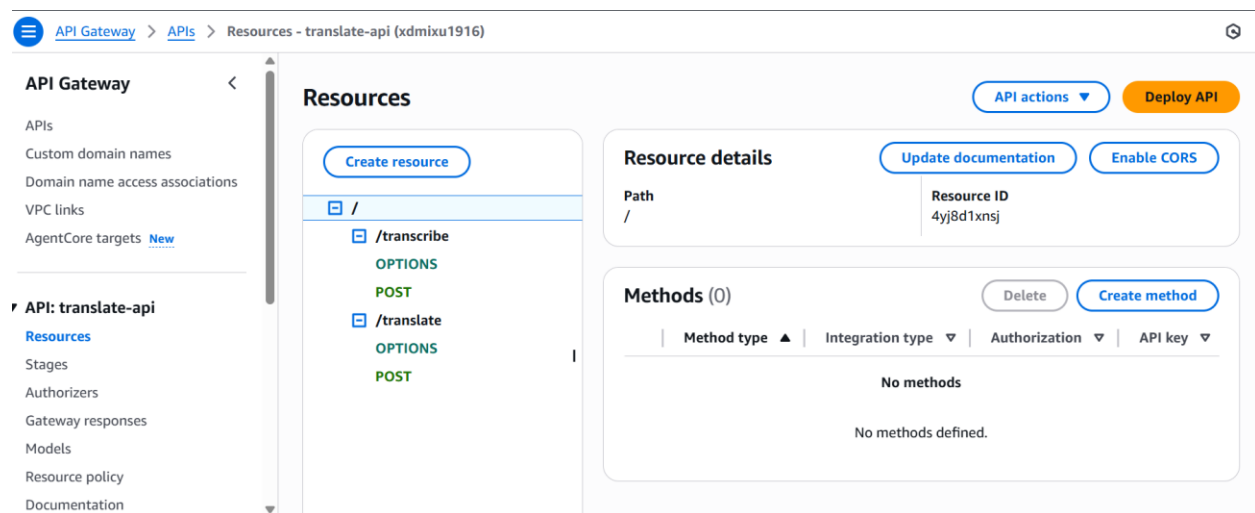
This solution is built entirely on serverless services, meaning costs scale based on usage and can remain very low for testing or low-traffic applications. AWS Lambda charges per millisecond of compute time, and typical transcription or translation workloads will cost only fractions of a dollar unless invoked at very high volume. Amazon Transcribe charges per minute of processed audio, with pricing based on the audio duration rather than transcription results. Amazon Translate charges per million characters processed, making short text translations inexpensive. S3 storage costs apply for storing audio files and transcription outputs, though typical storage usage for short audio clips is negligible. API Gateway charges per API call, making it affordable for small apps but potentially a cost factor for high-frequency workloads. Overall, the architecture is cost-efficient, but setting lifecycle policies for S3 storage, monitoring usage with CloudWatch and setting budget alerts is recommended.

8. Future Enhancements

Several improvements can enhance the system's functionality and user experience. One potential enhancement is integrating Amazon Comprehend to analyze the sentiment or extract key entities from the transcribed text. Another improvement is implementing automatic language detection before translation to reduce user input and streamline the workflow. For richer audio processing, you could replace the batch Transcribe job with Amazon Transcribe Streaming, allowing near real-time transcription with WebSocket connections. Adding authentication with Amazon Cognito would secure API endpoints and prevent unauthorized usage. Finally, the application could be expanded by introducing a database such as DynamoDB to store transcripts, translations, and user history for analytics and improved user experience.

9. Proof of Concept

Provision of API Gateway resources.



Full demo video [here](#)

S3 bucket showing uploaded audio in webm format.

The screenshot shows the Amazon S3 console interface for the bucket 'transcribe-audio-bucket-a2db0c61'. The 'Objects' tab is selected, displaying a list of three objects. The first two objects are audio files in webm format, both uploaded on December 6, 2025. The first object is 'audio-1764985000007.webm' (55.0 KB) and the second is 'audio-1764985106128.webm' (99.3 KB). Both are stored in the 'Standard' storage class. The third object is partially visible as 'audio-'.

Name	Type	Last modified	Size	Storage class
audio-1764985000007.webm	webm	December 6, 2025, 01:36:41 (UTC+00:00)	55.0 KB	Standard
audio-1764985106128.webm	webm	December 6, 2025, 01:38:27 (UTC+00:00)	99.3 KB	Standard
audio-		December 6, 2025,		

Provision of Lambda function and cloudwatch showing logs for debugging and monitoring.

The screenshot shows the AWS Lambda console for the function 'translate-function'. The 'Function overview' tab is selected, displaying a diagram of the function's configuration. The function is linked to an 'API Gateway' trigger. The 'Layers' section shows '(0)' layers. The 'Description' section provides details about the function, including its last modified time (12 minutes ago) and its ARN.

Function overview

Diagram | Template

translate-function

Layers (0)

API Gateway

+ Add trigger

+ Add destination

Description

-

Last modified

12 minutes ago

Function ARN

arn:aws:lambda:us-east-1:123456789012:function:translate-function

Function URL | Info

-

Code | Test | Monitor | Configuration | Aliases | Versions

Full demo video [here](#)

CloudWatch > Log management > /aws/lambda/translate-function > 2025/12/06/[\${LATEST}]0f53fd59a19443f4b6eb3f0ec8ebec39

CloudWatch

Favorites and recents

Dashboards

Alarms 0 0 0 0

In alarm

All alarms

Billing

AI Operations [New](#)

GenAI Observability

Application Signals (APM) [New](#)

Infrastructure Monitoring

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events - press enter to search

1m 1h UTC timezone

Display

Timestamp	Message
2025-12-06T01:39:30.989Z	f745ae1d-ff12-485a-8621-657c504f220e INFO Audio buffer size: 60188
2025-12-06T01:39:31.128Z	2025-12-06T01:39:31.128Z f745ae1d-ff12-485a-8621-657c504f220e INFO Successfully uploaded to S3
2025-12-06T01:39:31.128Z	f745ae1d-ff12-485a-8621-657c504f220e INFO Successfully uploaded to S3
2025-12-06T01:39:31.128Z	2025-12-06T01:39:31.128Z f745ae1d-ff12-485a-8621-657c504f220e INFO Starting transcription job..
2025-12-06T01:39:31.128Z	f745ae1d-ff12-485a-8621-657c504f220e INFO Starting transcription job: job-1764985171128

Back to top

CloudShell Feedback Console Mobile App

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudWatch > Log management > /aws/lambda/translate-function > 2025/12/06/[\${LATEST}]0f53fd59a19443f4b6eb3f0ec8ebec39

CloudWatch

Favorites and recents

Dashboards

Alarms 0 0 0 0

In alarm

All alarms

Billing

AI Operations [New](#)

GenAI Observability

Application Signals (APM) [New](#)

Infrastructure Monitoring

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events - press enter to search

1m 1h UTC timezone

Display

Timestamp	Message
2025-12-06T01:39:45.802Z	2025-12-06T01:39:45.802Z f745ae1d-ff12-485a-8621-657c504f220e INFO Extracted transcription: W...
2025-12-06T01:39:45.802Z	f745ae1d-ff12-485a-8621-657c504f220e INFO Extracted transcription: We will get there.
2025-12-06T01:39:45.803Z	END RequestId: f745ae1d-ff12-485a-8621-657c504f220e
2025-12-06T01:39:45.803Z	END RequestId: f745ae1d-ff12-485a-8621-657c504f220e
2025-12-06T01:39:45.803Z	REPORT RequestId: f745ae1d-ff12-485a-8621-657c504f220e Duration: 14839.02 ms Billed Duration: 14839.02 ms

Back to top

CloudShell Feedback Console Mobile App

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Full demo video [here](#)

CloudWatch > Log management > /aws/lambda/translate-function > 2025/12/06/[\${LATEST}]0f53fd59a19443f4b6eb3f0ec8ebec39

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events - press enter to search

Clear 1m 30m 1h 12h Custom UTC timezone

Display

Timestamp	Message
2025-12-06T01:39:39.526Z	2025-12-06T01:39:39.526Z f745ae1d-ff12-485a-8621-657c504f220e INFO Attempt 4: Job status = IN_PROGRESS
2025-12-06T01:39:41.573Z	2025-12-06T01:39:41.573Z f745ae1d-ff12-485a-8621-657c504f220e INFO Attempt 5: Job status = IN_PROGRESS
2025-12-06T01:39:43.629Z	2025-12-06T01:39:43.629Z f745ae1d-ff12-485a-8621-657c504f220e INFO Attempt 6: Job status = IN_PROGRESS
2025-12-06T01:39:43.629Z	f745ae1d-ff12-485a-8621-657c504f220e INFO Attempt 6: Job status = IN_PROGRESS
2025-12-06T01:39:45.679Z	2025-12-06T01:39:45.679Z f745ae1d-ff12-485a-8621-657c504f220e INFO Attempt 7: Job status = COMPLETED
2025-12-06T01:39:45.679Z	f745ae1d-ff12-485a-8621-657c504f220e INFO Attempt 7: Job status = COMPLETED

Back to top

CloudShell Feedback Console Mobile App © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Full demo video [here](#)

Provision of Transcribe resource showing various job IDs and transcribed data.

Amazon Transcribe

Real-time transcription
Transcription jobs
Customised language model
Customised vocabulary
Vocabulary filtering

Amazon Transcribe Call Analytics

Real-time analytics
Post-call analytics
Category management

Amazon Transcribe Medical

Real-time transcription
Transcription jobs

Amazon Transcribe > Transcription jobs

Transcription jobs (3) Info

Download Copy Delete Create job

Find job names Status: All

Name	Status	Language	Language settings	Model type	Model
job-1764985171128	Complete	English, US (en-US)	Specific language	General	-
job-1764985106452	Complete	English, US (en-US)	Specific language	General	-
job-1764985001066	Complete	English, US (en-US)	Specific language	General	-

aws

Search

[Alt+S]

1

United States (N. Virginia)

Acc

Amazon Transcribe

Real-time transcription

Transcription jobs

Customised language model

Customised vocabulary

Vocabulary filtering

Amazon Transcribe Call Analytics

Real-time analytics

Post-call analytics

Category management

Amazon Transcribe Medical

Real-time transcription

Transcription jobs

Expiry

Info

The transcription is available for 89 more days.

Input file format

webm

Audio sampling rate

48000 Hz

Vocabulary filter

-

Toxicity detection

-

Transcription preview

Select download to save a local copy of the transcription.

Down

Text

Audio identification

Subtitles

Toxicity detection - new

We will get there.

CloudShell

Feedback

Console Mobile App

© 2025, Amazon Web Services, Inc. or its affiliates.

Privacy

T

Full demo video [here](#)