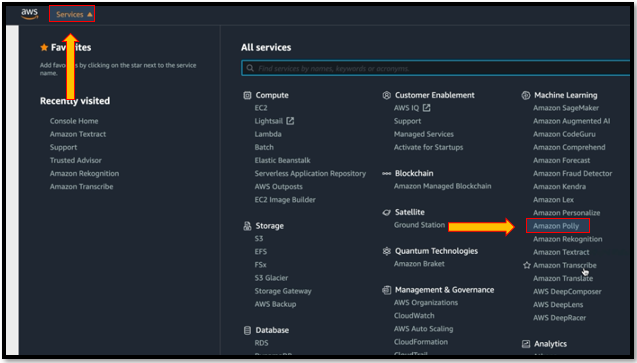
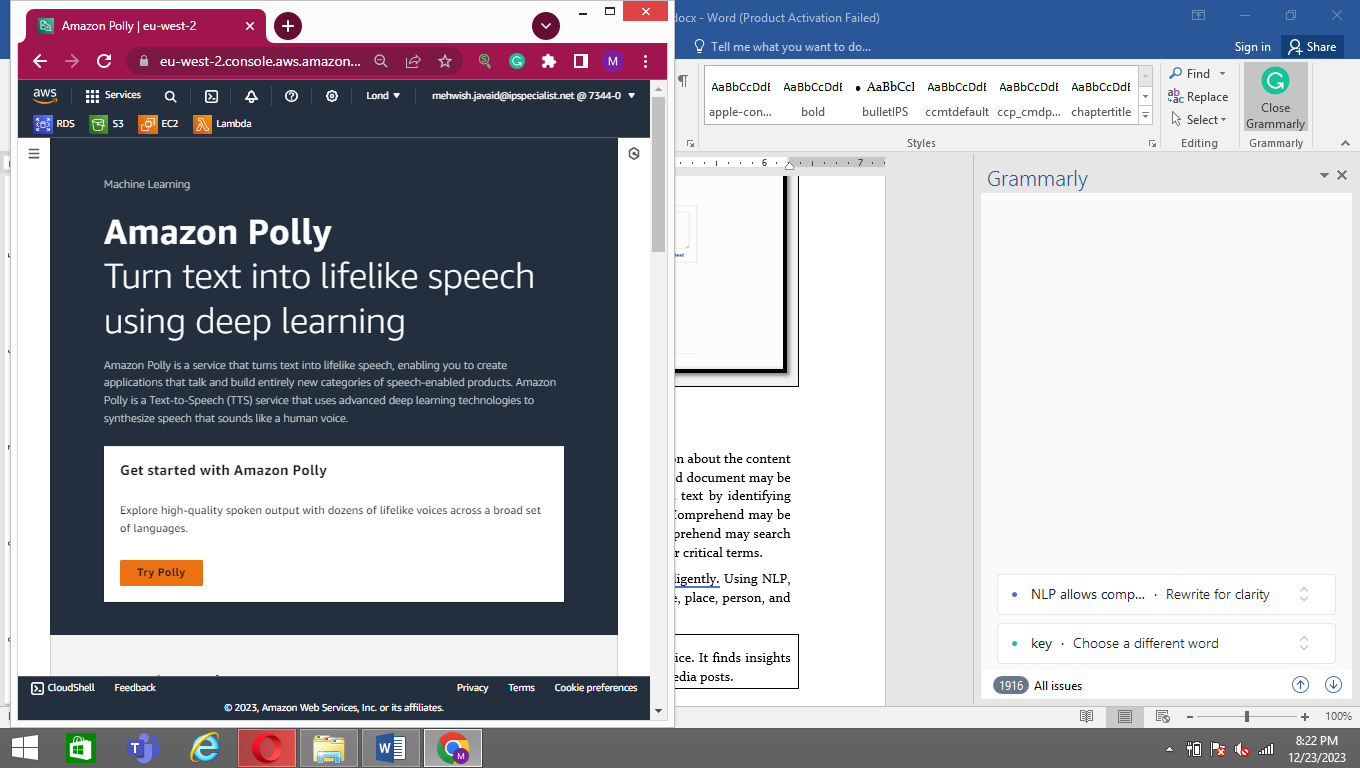
## Demo 5-01: Amazon Polly

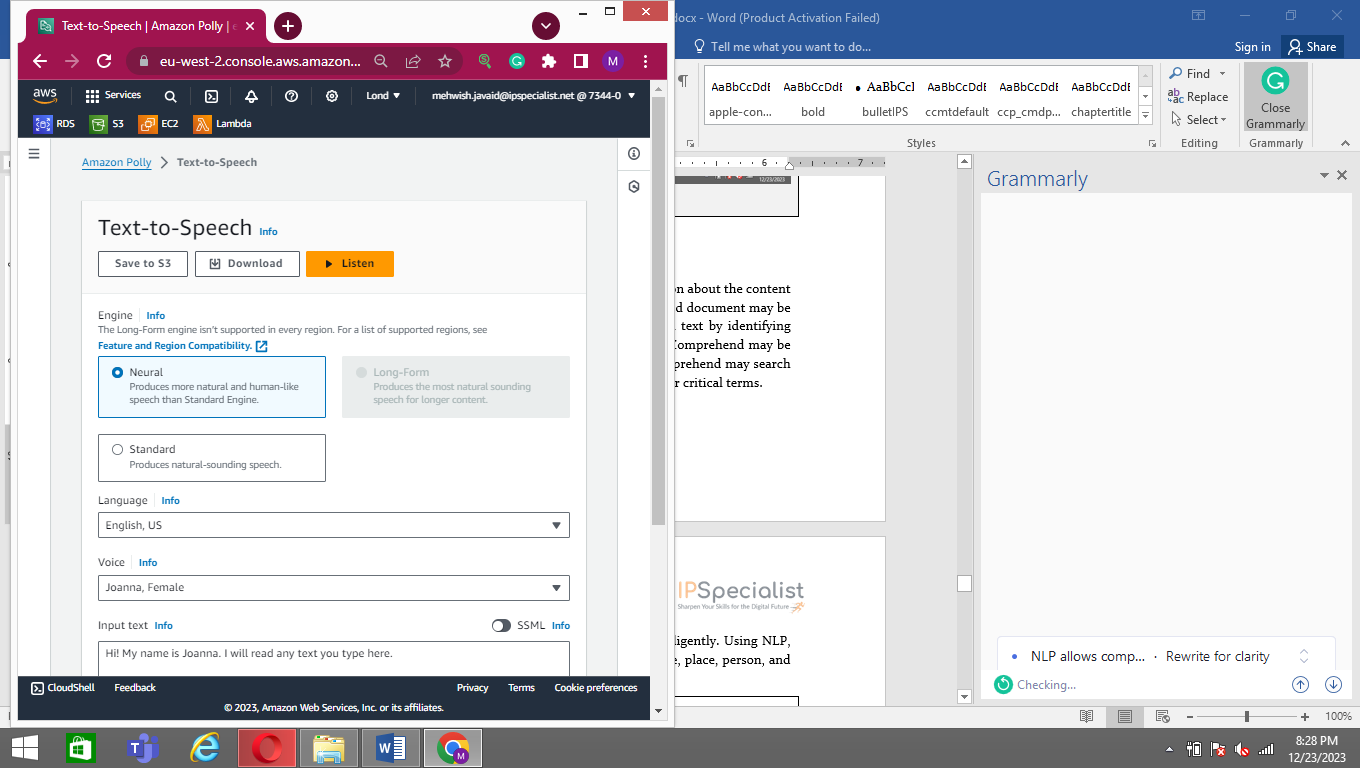
1. Log into the AWS console.
2. Go to “Services”.
3. Go to “Machine Learning”.
4. Click on “Amazon Polly”.



5. Clck on “Try Polly”.



6. Here, you can write the text, select the language and region, select a voice to listen to the speech, and even download it.



## Lab 5-01: Use Amazon Rekognition

### Lab Prerequisites

* Familiarity with basic AWS Cloud Computing concepts and terminology
* An AWS account with an active subscription

### Service Introduction

Amazon Rekognition is a way of converting images into tags or texts. It is a way of allowing your software to see the image. You can upload an image, and Rekognition will tell you what it thinks the image is with a certain degree of confidence. It can be used with lots of apps.

You can recognize objects, scenes, people, activities, and language in photographs and videos and detect inappropriate content using Amazon Rekognition. You may utilize Amazon Rekognition's highly accurate facial analysis and facial search capabilities to recognize, analyze, and compare faces for user verification, people counting, and public safety use cases.

### Case Study Enterprise Healthcare - Cleveland Medical Center

Background

Cleveland Medical Centers are stretched across the three states of California, Oregon, and Hawaii. The centers serve millions of lives through hospital admissions, emergency, clinic, outpatient, and home care visits. Cleveland uses advanced technology that helps restore patient health and empowers physicians to deliver better care to achieve better outcomes.

Cleveland is currently in the middle of a digital transformation to bring patient care closer to home. They aim to treat patients at the nearest hospital or resource center in their area or dispatch healthcare workers to assist them in their homes. To do this, they plan to use video conferencing, IoT, remote monitoring, and mobile solutions to enable people to get the healthcare they need, thus limiting the number of patients traveling to one of their hospitals. This way is not only cost-effective, but it also gives patients a better experience.

An external consultant has done an extensive study on the Cleveland infrastructure. The consultant has advised the board to build a “Cloud First” strategy. It will reduce the need for organizations to invest in and maintain their on-premises IT resources, scalability to meet workload and user demands, and fewer wasted resources because the company will only pay for what they use.

Business Challenge

Cleveland's business development team has deployed an online appointment portal where users conveniently schedule appointments. Hundreds of patients come to the medical centers daily through online appointments, and management has faced trouble identifying the patients by appointment number only. Cleveland's business development team wants a solution that identifies the patient by face and shows the basic details. The management team hired a Cloud Specialist to propose a solution for the current requirement.

Proposed Solution

You are the Cloud Specialist for Cleveland medical center. You proposed a solution to this requirement by using Amazon Rekognition. Cleveland's management team can easily identify the patient using the facial analysis feature of Amazon Rekognition. To use this feature, the picture of the patients should be uploaded at the time of the appointment. When patients come for medical consultancy, management identifies the patient through the Amazon Rekognition service.

Lab Diagram

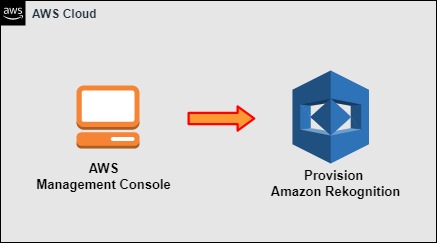


Figure 5-02: Amazon Rekognition

Implementation Steps

1. Navigate to the Amazon Rekognition.
2. Use Facial Analysis.

### Solution

|  |
| --- |
| **Step 1: Navigate to the Amazon Rekognition**   1. Log in to **AWS Management Console**. 2. Click on **Services**. 3. Click on **Amazon Rekognition** under **Machine Learning**.      1. On the **Amazon Rekognition** page, click on the **Facial analysis** feature.     **Step 2: Use Facial Analysis**   1. When the **Facial analysis** page appears, scroll down and click on **Upload** to upload the patient’s picture.        1. The facial analysis will show the **Result** after analyzing the **picture**. 2. The **result** will be shown below. |

## Lab 5-02: Creating a Lambda Function

### Lab Prerequisites

* Familiarity with basic AWS Cloud Computing concepts and terminology
* An AWS account with an active subscription

### Service Introduction

AWS Lambda is a serverless compute service that lets you run code without needing to deploy or manage servers, design workload-aware cluster scaling logic, maintain event integrations, or manage runtimes. You can run code for nearly any application or backend service with Lambda, and you do not have to worry about administration. Therefore, you do not need to worry about which AWS resources to launch or how well you manage them. Instead, you must upload the code to Lambda, and it will execute.

The code in AWS Lambda is performed in reaction to events in AWS services, such as adding/deleting files in an S3 bucket, making an HTTP call to the Amazon API gateway, and so on. On the other hand, Amazon Lambda can only be utilized to do background activities.

### Case Study Enterprise Healthcare - Cleveland Medical Center

Background

Cleveland Medical Centers are stretched across the three states of California, Oregon, and Hawaii. The centers serve millions of lives through hospital admissions, emergency, clinic, outpatient, and home care visits. Cleveland uses advanced technology that helps restore patient health and empowers physicians to deliver better care to achieve better outcomes.

Cleveland is currently in the middle of a digital transformation to bring patient care closer to home. They aim to treat patients at the nearest hospital or resource center in their area or dispatch healthcare workers to assist them in their homes. To do this, they plan to use video conferencing, IoT, remote monitoring, and mobile solutions to enable people to get the healthcare they need, thus limiting the number of patients traveling to one of their hospitals. This way is not only cost-effective, but it also gives patients a better experience.

An external consultant has done an extensive study on the Cleveland infrastructure. The consultant has advised the board to build a “Cloud First” strategy. It will reduce the need for organizations to invest in and maintain their on-premises IT resources, scalability to meet workload and user demands, and fewer wasted resources because the company will only pay for what they use.

Business Challenge

Cleveland's business development team plans to shift its resources from on-premises to the cloud. One of the employees suggested using AWS cloud service providers to pursue this requirement. After migration, Cleveland wants automation for managing the underlying compute service. The management team hired a Cloud Specialist to propose a solution for the current requirement.

Proposed Solution

You are the Cloud Specialist for Cleveland medical center. You proposed a solution to this requirement by using AWS Lambda, and Cleveland can automatically manage the underlying compute resources for you with high scalability, performance, and security.

Lab Diagram

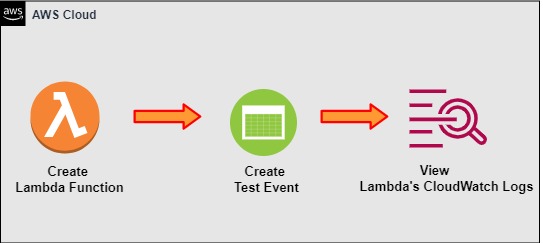


Figure 5-04: Lambda Function

Implementation Steps

1. Author Node.js Lambda Function in AWS Console.
2. Test Function Using a Test Event.
3. View Lambda's CloudWatch Logs.

### Solution

|  |
| --- |
| **Step 1: Author Node.js Lambda Function in AWS Console**   1. Log in to **AWS Management Console**.      1. Click on **Services**. 2. Click on **Lambda** under **Compute**.      1. Click on **Create function**.      1. Select the **Author from scratch**.      1. Set the following **values**:   Function name: **AWSURLChecker**  Runtime: **Node.js**   1. Expand **Change default execution role**. 2. For the **Execution role**, select **Create a new role with basic Lambda permissions**. 3. Click **Create function**.      1. The function is now successfully created.      1. Now, click on **Code**. Click on **index.js** and **delete** the code.      1. Then, go to the **GitHub link** provided and copy the code.   <https://github.com/linuxacademy/content-aws-ccp-labs/blob/master/creating-a-lambda-function-using-the-aws-console/index.js>     1. Paste it in **index.js** and deploy.     **Step 2: Test Function Using a Test Event**   1. Click **Test** in the upper right. 2. For the **Event** template, enter **AWSTestEvent**. 3. Leave the other **default** values. 4. Click **Save changes**.      1. When the **Test event** is saved, click on **Test**.      1. Review the **output response**. 2. It should be **200**, which means the website is **up** and **running**.     **Step 3: View Lambda's CloudWatch Logs**   1. Click the **Monitoring** tab. 2. Click **View logs** in **CloudWatch**.      1. Click on the most recent log stream in the **Log streams** section.        1. Notice the **Billed duration**. |