Omnidya Dashcam- Facial Recognition

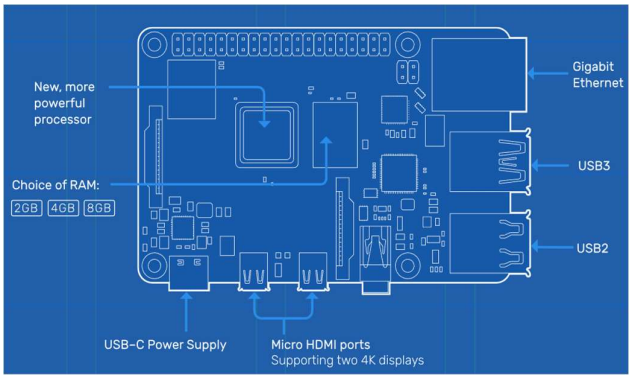
Dhruv Patel, Malav Sukhadia, Harsh Viradia

Mentor: Madhav Ajwalia & Hemant Yadav

IT Department, Charotar University of Science and Technology

Anand, Gujarat

Abstract: The Umbrella Project Aim, to create a mobile accessible application in Android + iOS which is Omnidya, it is advancing the insurance industry by leveraging emerging technologies to create a comprehensive, data-driven solution that is easy to use.



The Internship Aim, to build a working flow of capturing image from a dashcam in a user car and to recognize face through Amazon Web Services, Validate & Verification that face is Driver or Any other registered person on Application and Store the Data.

Technology Utilization:

Amazon Web Services: Amazon Web Services (AWS) is the most comprehensive and maximum used cloud platform, providing more then 200 comprehensive data center services. many of customers which include startups, large comaney and major goverment agecies, use AWS to reduce costs and become more agile, and inovate faster.

AWS the most comprehensive global cloud infrastructure. Rather then cloud service provider offers low latency, high throughput, and multiple Availability Zones associated with maximum redundant networks. AWS has 84 AZs in 26 gregions around the world, And it announced plans for an additional 24 AZs and 8 additional AWS Zones in Australia, Canada, India, Israel, New Zealand, Spain, Switzerland and the United Arab Emirates (UAE). , The AWS Region and Availability Zone model is recognized by Gartner as the recommended approach for runing entrprise aplictions that require higvailability.

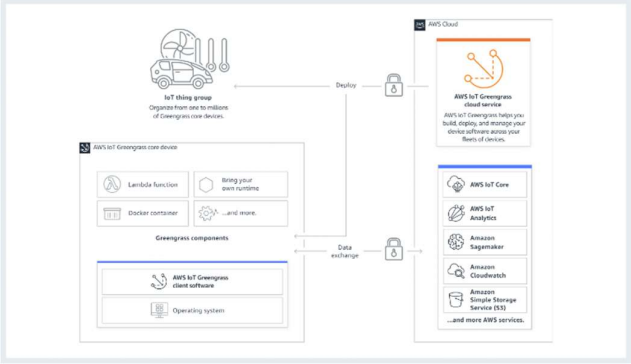
AWS consists of many Cloud Services which is being used during the internship. AWS Products used are mentioned below: -

- S3 (Simple Storage Service)
- Lambda Function
- API Gateway
- Triggers
- SNS

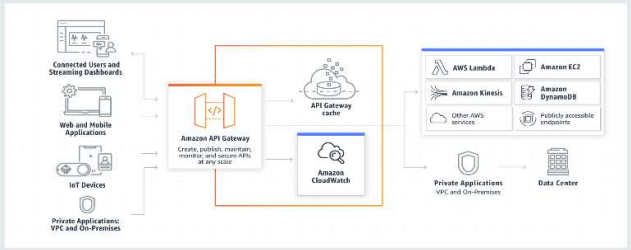
- IoT Greengrass

AWS Lambda: AWS Lambda is an event-based server les service that run code for virtually any type of backend application/service without managing a server. You can activate Lambda as a Service (SaaS) applications from over 200 services and software and only pay for what you use.

AWS IoT Greengrass: AWS IoT Greengrass Client software, also known as AWS IoT Greengrass Core software, runs on Windows and Linux-based distributions for devices with ARM or x86 architectures, such as Ubuntu or Raspberry Pi. With AWS IoT GreenGrass, you can program devices to run smoothly when built, run Machine learning models, filters and predictions based on aggregated device data. AWS IoT Greengrass enables local implementation of Lambda functions, Docker containers, native OS-processes, / custom runtimes of your choice.



POSTMAN: Postman is an API platform for building and using APIs. Postman simplifies each stage of the API lifecycle and streamlines collaboration so you can build better APIs, faster. The Postman platform includes a comprehensive set of tools that help accelerate the API lifecycle from design, test, documentation, and mock sharing to more Discoverability of your API. Postman Workplace helps you organize your work with APIs and



collaborate in your organization or around the world. There are three different types of postman workspaces for your different needs: personal workspaces, group workspaces, and public workspaces.

AWS Lambda Functions: Lambda Functions are Middle nodes which are used to integrate and build a autonomous working AWS Flow. Lambda Functions are set of instructions which are triggered or run to perform specific functions accessing different Amazon Web Services. Lambda Function Used in Application flow consist of 2 Lambda Functions.

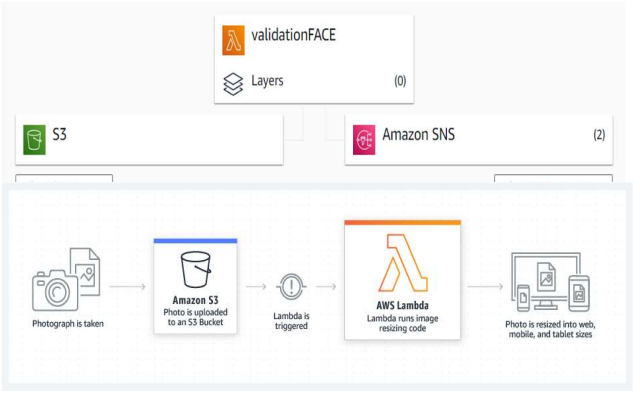
1. API to S3 Lambda Function
2. S3 Trigger Lambda Function

API to S3 Lambda Function: When AWS API is called by the Application through POST API with Binary Image attached to it. This function Convert the binary image to Image Object. And

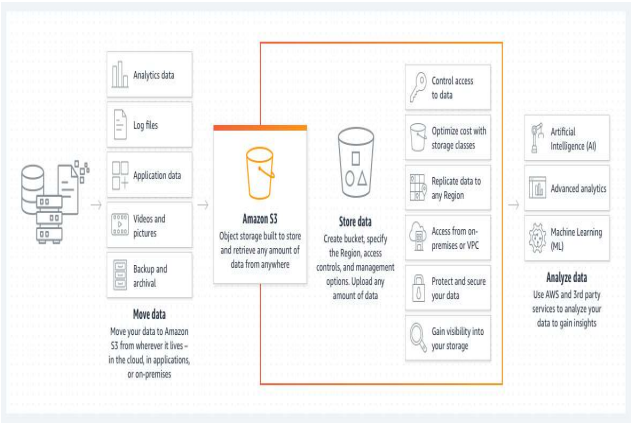
It will Validate the Image according to Face

Validation Criteria.

S3 Verification Lambda Function: When any weight/ File will be added in S3 Bucket this Lambda Function gets triggered and will Compare Faces from added image with existing image proof's and if it is verified then it will be added to collection.



AWS S3 Bucket: Utilization of S3 bucket is to Store the Files (Images) received by AWS API POST request and to be a Storage unit which triggers another AWS Lambda Function as Trigger. It is easy to access Service. We have demonstrated different POST protocol which directly adds files to AWS S3.



It has another Feature of generating Pre-signed URL. Which is a unique URL address which is used to access the file from anywhere using the URL.

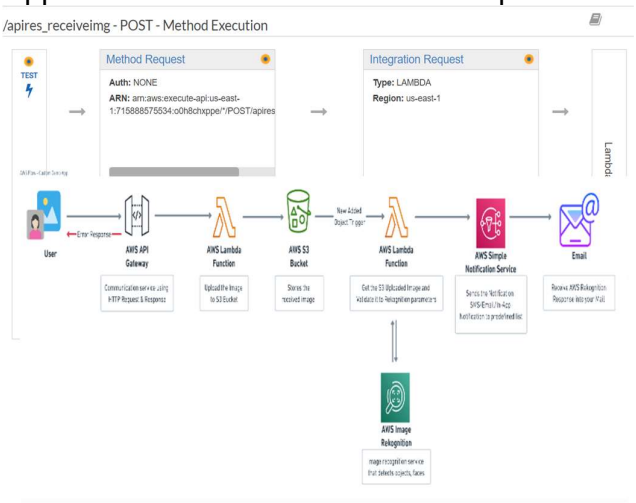
AWS API Gateway: API Gate way is a key component in exchanging information with S3 Bucket and Lambda function. We cannot access the S3 Bucket Directly because it may create a security issues if we share secret key to access bucket with everyone. So, API Gateway is used to interact with AWS Cloud Features. Flow is to Send a POST request with Header and Image in Binary format inside the Request Body. It is send using the API Link/Path which is publicly available. It is send to AWS API Gateway which identify the Image in the body and decodes the POST Request and Provides a JSON Format Object to Triggering/ receiving Lambda function. And can be directly accessed with Object Calling in Lambda Function.

It is also useful while any Error occurs in Uploading. We can send response status code with respective error code. Which can be used in application to handle such occurrences.

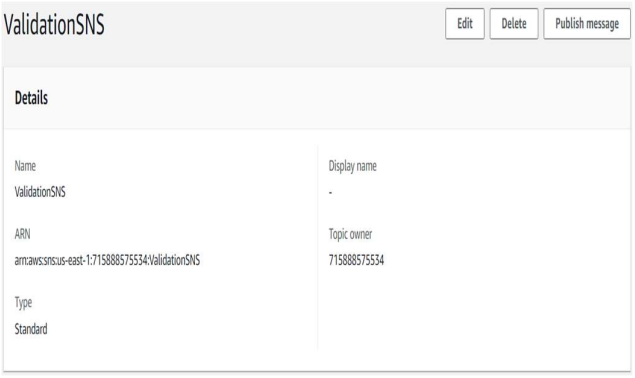
AWS SNS: AWS SNS i.e. Simple Notification Service which is used to send a notification/ message to pre-set/particular predefined recipient. It provides SMS, Email, In-App Notification service under SNS. In the Demo Flutter Application, When Verification Lambda function verifies the face with existing data set, it then triggers a SNS Request with AWS Rekognition Response Object which includes recognised features by AWS Rekognistion.

Flutter Application: Flutter is an open source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase. Flutter is used to build Mobile Application using Dart Language.

Basic Application: A Flutter Application is made to demonstrate the similar flow which is being implemented in Original Company Application which is underdevelopment. It



includes Image Name Field, Single Image Picker, Text Area which shows the HTTP POST Request Response Status Code and Status Error.



API Integration: Main Aim of the Application includes to transfer of image/video data from Flutter Mobile Application to AWS S3 Cloud, which is performed by API Gateway and Flutter Package. We have demonstrated and tested the Upload Data flow and have received the Response Message from successful Upload.

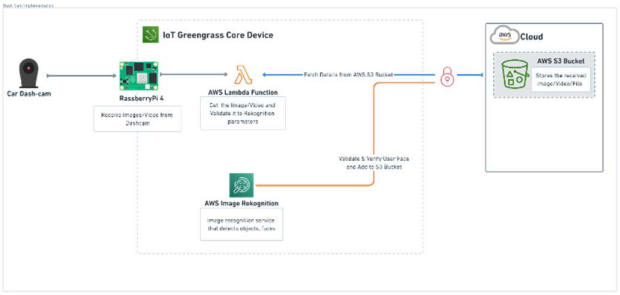
```
String bucketname='reciver-bkt';
String uploadIMGname='TRIAL_PUT1.jpg';
String StageURL='https://rzdgmqv4f2.execute-api.us-east-1.amazonaws.com/v1/';

var uri = Uri.parse(StageURL+bucketname+'/'+uploadIMGname);
final http.Response response = await http.put(uri,
    headers: {'testsource':'testtoken','Content-Type': 'image/jpg','Accept': 'image/jpg'},body: image!.readAsBytesSync());

var headers = {
  'testsource': 'testtoken',
  'Content-Type': 'image/jpg'
};
```

Application Working Flow: The Flow in which the application is working is included in this chapter. Original Application “Omnidya : Dashcam Feature” which is underproduction was internal asset of Company which cannot be revealed but have demonstrated a demo application which is performing the same functionality and can be shown without any restrictions.

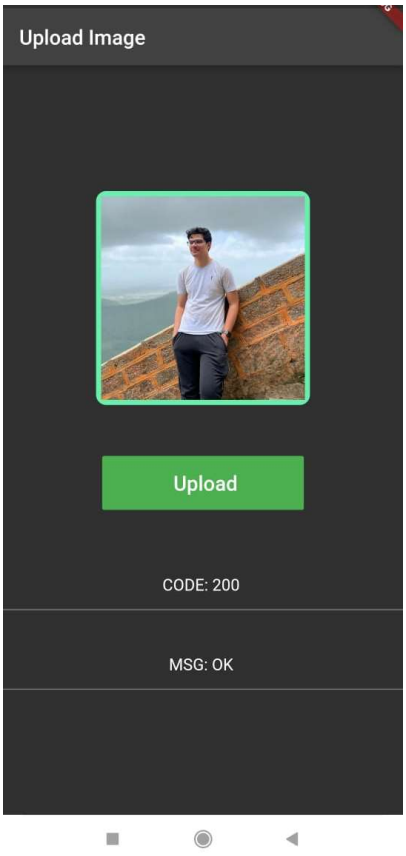
AWS Usage Flow: Amazon Web Services is used to demonstrated whole flow of API Request receiving to Image Processing and Recognition to replying with API RESPONSE to Flutter Application used to display it. This Flow is implemented in Demo Project which is allowed to show outside the Company.

Application Data Working Flow: Flow of Main Application is demonstrated below, First Camera Captures Image, It Transmits/processes the Data using IoT Greengrass Core Device facilities of AWS to process AWS Rekognistion in IoT Device like Raspberry PI 4 or any other device. After Processing it will Ping AWS API Gateway and trigger a Lambda Function which will store the Verified and Validated Faces in AWS S3 Cloud Bucket.

REFERENCE:



1. https://aws.amazon.com/what-is-aws/



2. https://aws.amazon.com/s3/
3. https://aws.amazon.com/lambda/
4. https://aws.amazon.com/api-gateway/
5. https://aws.amazon.com/sns/
6. https://aws.amazon.com/greengrass/
7. https://docs.aws.amazon.com/greengrass/v2/developerguide/how-it-works.html
8. https://www.postman.com/product/what-is-postman/
9. https://www.netguru.com/blog/is-flutter-a-programming-language
10. https://flutter.dev/