

# 雲端架構技術的理論與實踐

企業架構科 實習生 吳旻聖(Akira)/高聖哲(Jerry)



# 目錄(TABLE OF CONTENTS)

1

自我介紹

2

**RD技術研究**

- 技術趨勢學習
- Gartner文章閱讀
- RD分享會

3

**專案內容**

- 專案發想
- 使用情境
- 軟體架構流程
- Demo

4

心得分享



# 自我介紹

名字:高聖哲(Jerry)

學校:清華大學

系所:通訊工程所(升碩一)



國泰金控 企業架構科 實習生



# 自我介紹

名字:吳旻聖(Akira)

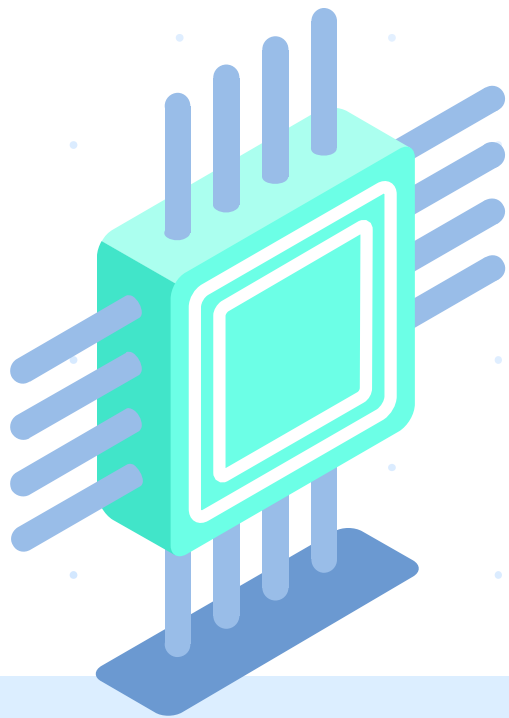
學校:清華大學

系所:電機工程系碩士班



國泰金控 企業架構科 實習生





# RD技術研究分享

Microservice 、 Serverless 、 Service Mesh 、 APM

1

# 技術學習

4個技術

## 趨勢文章閱讀

8篇Gartner文章

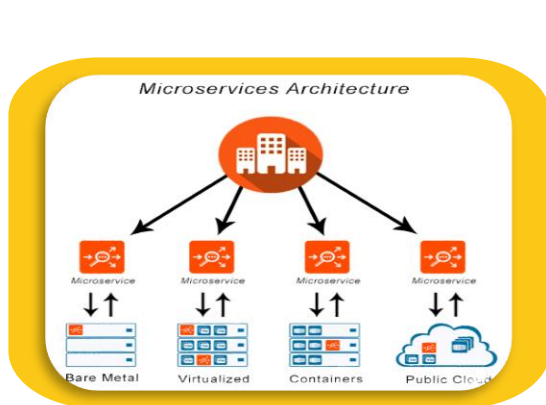
## RD技術分享會

2場分享會



# 技術研究與學習

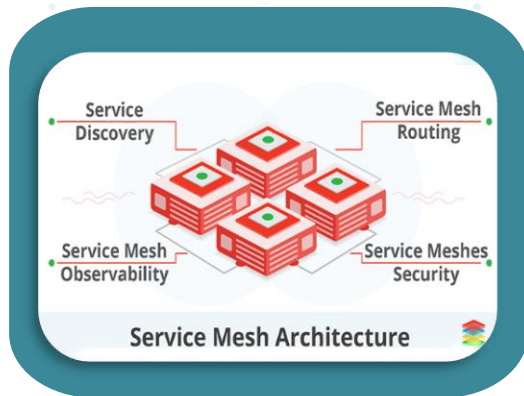
Microservice



Serverless



Service Mesh



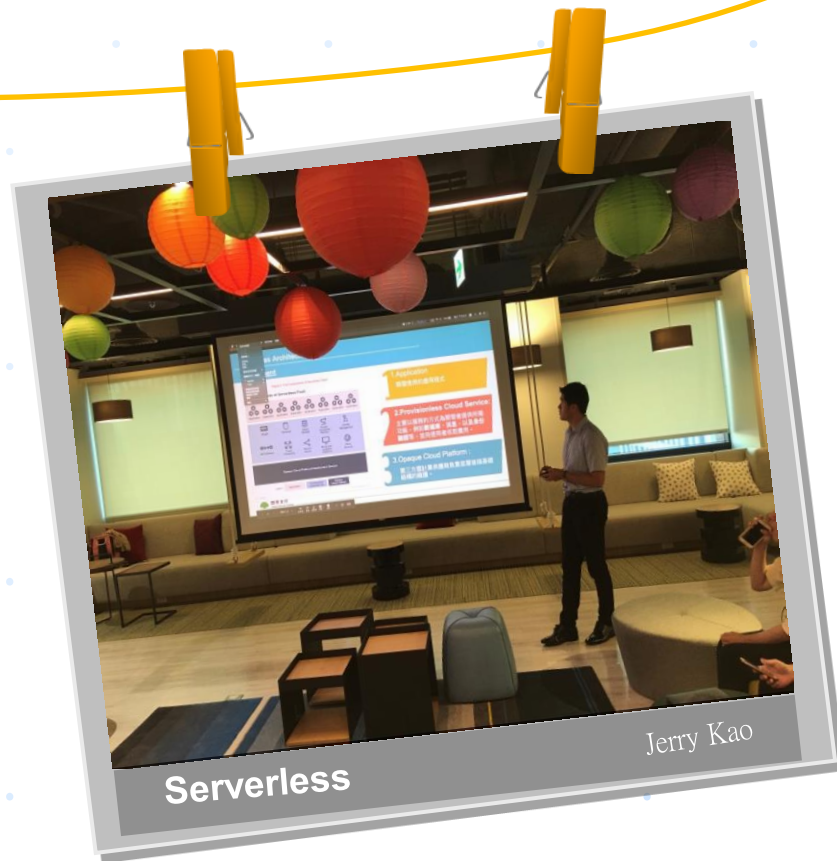
APM



[illegible]



# RD技術分享會



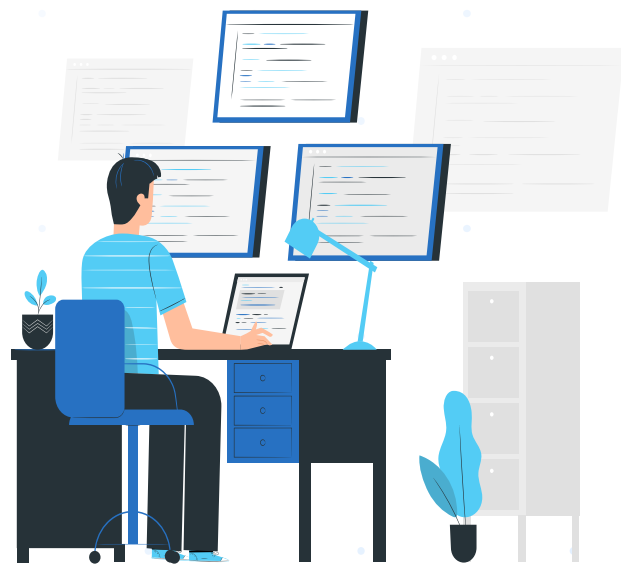
# 2



## 實習專案內容

Lambda應用OpenCV去背軟體

# 專案發想



## 使用情境



# Lambda 簡介



- 是AWS上一種無伺服器（Serverless）運算服務,執程式碼時無需佈建或管理伺服器
- 設定好觸發程式的條件，平台才會啟動運程式並收取費用, 減少額外資源浪費
- 支援 Java、Go、PowerShell、Node.js、C#、Python 和 Ruby 程式碼



# Lambda 運行三元素



設置

根據會使用到的  
套件進行layer  
的增加



演算法

欲在Lambda上  
運行的演算法



trigger

Lambda  
執行條件

# Lambda 運行環境設置



- Lambda 不支持線上安裝 package

- 需要使用創建 layer 的方式管理 package

**OpenCV**

一個跨平台的  
電腦視覺庫



**Scipy**

演算法庫和數學  
工具包



# Lambda 運行環境設置

Lambda 是利用新增 Layer 的方式來處理會使用到的套件，因此會由以下步驟進行



使用EC2安裝  
套件並打包成  
package



設定IAM role權  
限允許S3權限



將Package儲  
存至放置S3中



建立lambda函數  
將S3 package  
引入 layer層





# Lambda 運行環境設置

## Step1



```
ubuntu@ip-172-31-44-92:~$ mkdir -p build/python/lib/python3.6/site-packages
ubuntu@ip-172-31-44-92:~$ pip3 install opencv-python -t build/python/lib/python3
.6/site-packages
Collecting opencv-python
  Downloading https://files.pythonhosted.org/packages/37/49/874d119948a5a084a7eb
e98308214098ef3471d76ab74200f9800efeef15/opencv_python-4.0.0.21-cp36m-manyl
inux1_x86_64.whl (25.4MB)
```

使用EC2，並在環境中建立需要使用的package,同時安裝所需要的套件 (opencv)



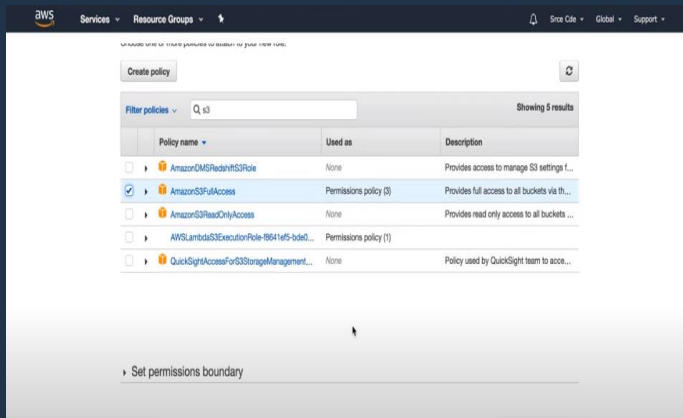
```
ubuntu@ip-172-31-44-92:~/build$ ls
package.zip  python
ubuntu@ip-172-31-44-92:~/build$
```

將package打包成壓縮檔

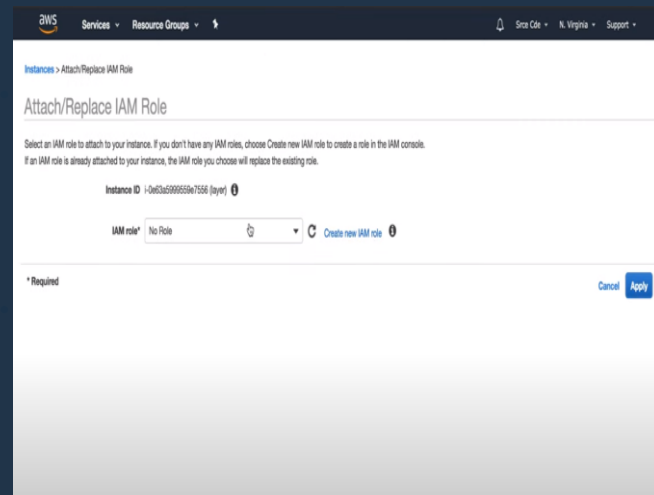


# Lambda 運行環境設置

## Step2



建立一個IAM role並需要新增  
AmazonS3FullAccess的權限



將剛剛所建立的IAM新增至  
EC2當中

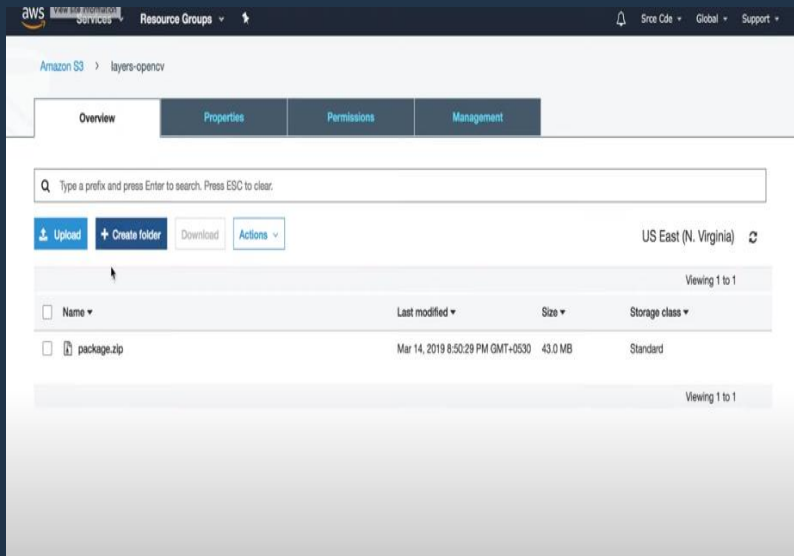


# Lambda 運行環境設置

## Step3



```
ubuntu@ip-172-31-44-92:~/build$ aws s3 cp package.zip s3://layers-opencv  
upload: ./package.zip to s3://layers-opencv/package.zip
```



擁有IAM role S3權限.後  
Package.zip檔即可新增  
至S3儲存空間



# Lambda 運行環境設置

## Step4



Layer configuration

Name

Description

Code entry type

For files larger than 10 MB, consider uploading using Amazon S3.

Compatible runtimes [Info](#)  
Choose up to 5 runtimes.

License [Info](#)

在Lambda layer設定中上傳剛剛S3當中的Package

Lambda > Layers > opencv-layer ARN: arn:aws:lambda:us-east-1:052674914236:layer:opencv-layer:1

opencv-layer

Successfully created layer opencv-layer version 1.

Version details

Version	Description	Created	License
1		now	

Compatible runtimes

python3.6

Layer層即新增完成,在撰寫Lambda function即可引入需要的套件



# Lambda 運行三元素



設置

根據會使用到的  
套件進行layer  
的增加



演算法

欲在Lambda上  
運行的演算法



trigger

Lambda  
執行條件

# 三大去背演算法

使用兩張圖片進行去背  
(人像 + 純背景)

透過對比分析背景

## 缺點

情境不適用、麻煩

## Alpha

透過畫出背景與人的區域去背

## 缺點

需要在前端做一個畫背景的物件，時間不太夠

## Deep Learning

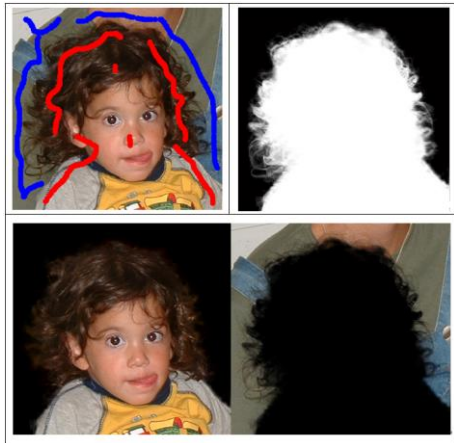
類似透過兩張照片，  
但可以支援到影片

## 缺點

資源用量極大  
不太適用於無伺服器服務

# 最後選擇

使用Alpha去背法，但因為開發時間的關係，使用固定的遮罩來進行去背



Alpha去背法示意圖



固定遮罩



# Lambda 運行三元素



設置

根據會使用到的  
套件進行layer  
的增加



演算法

欲在Lambda上  
運行的演算法



trigger

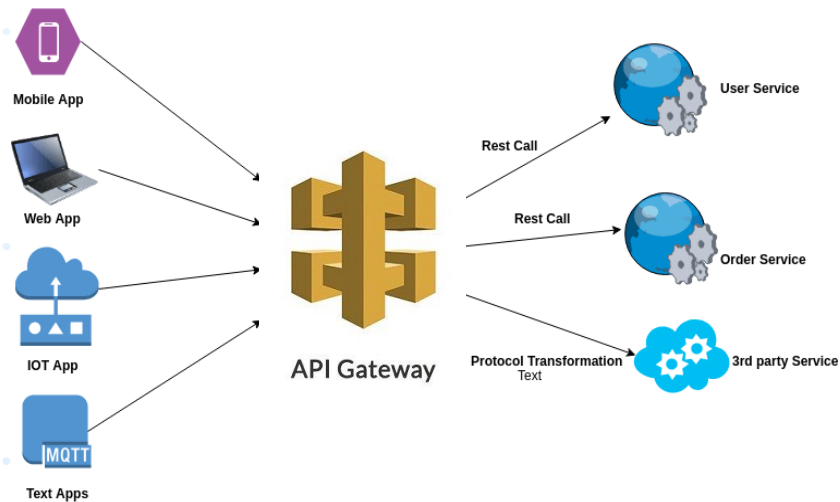
Lambda  
執行條件



# API Gateway 簡介

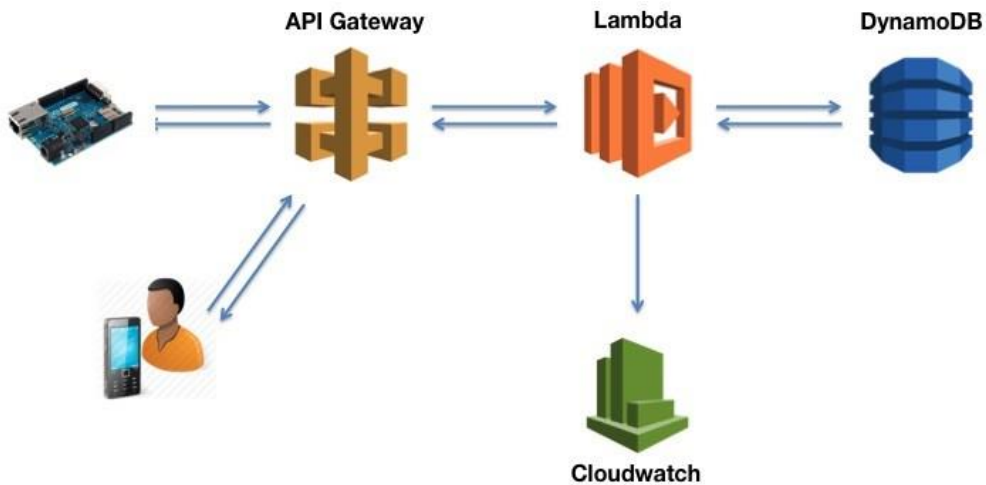


- 是一個伺服器
- 負責接收請求、轉發
- 可以提供api給使用者做使用

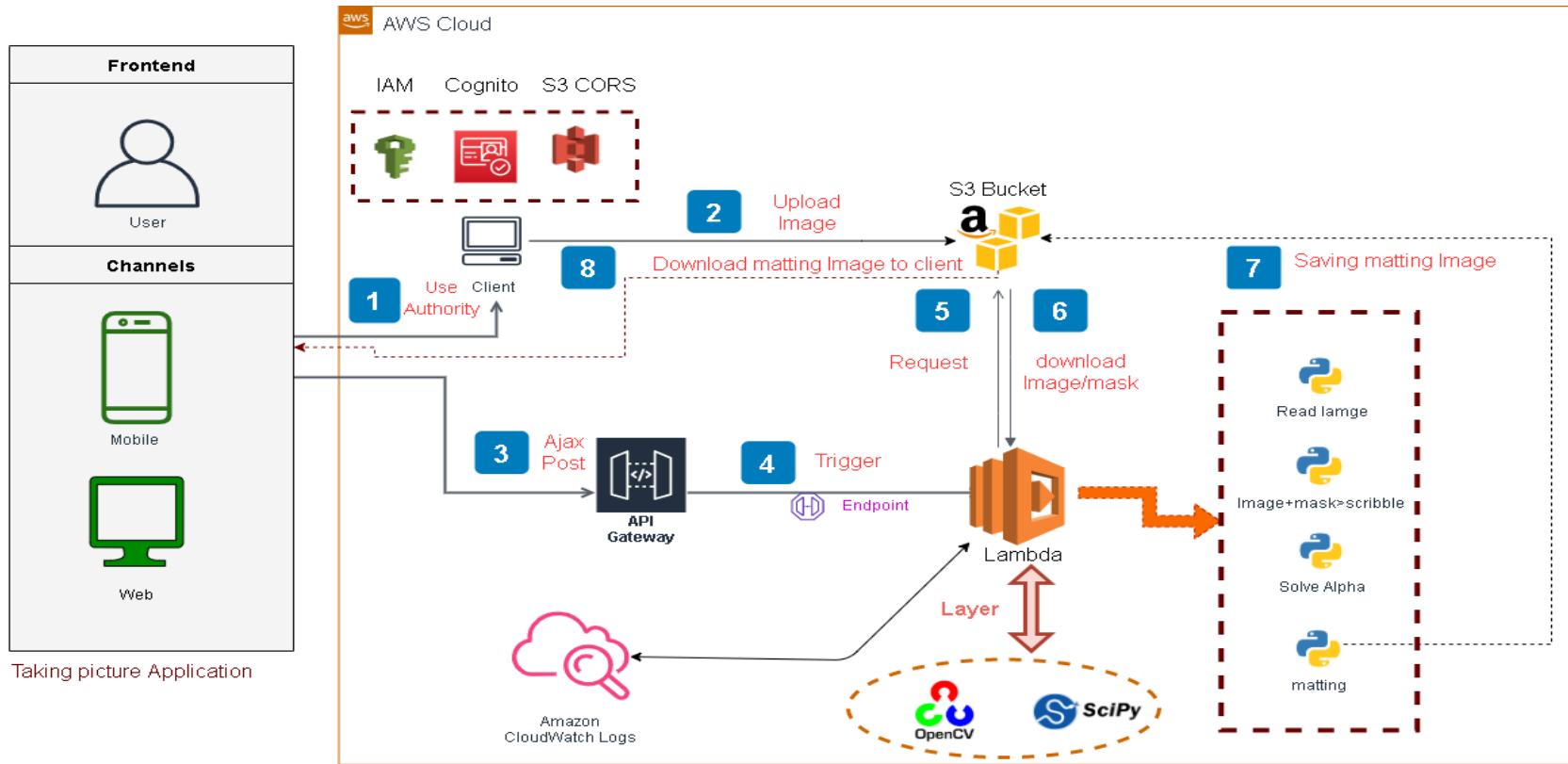


# Big Picture

用戶的請求會先進到API Gateway  
後再發到需要呼叫的服務



# 軟體架構圖-version1



# 流程1 Frontend

## Frontend

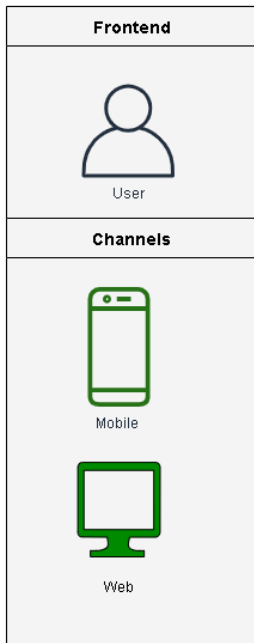
步驟1&2

步驟3&4

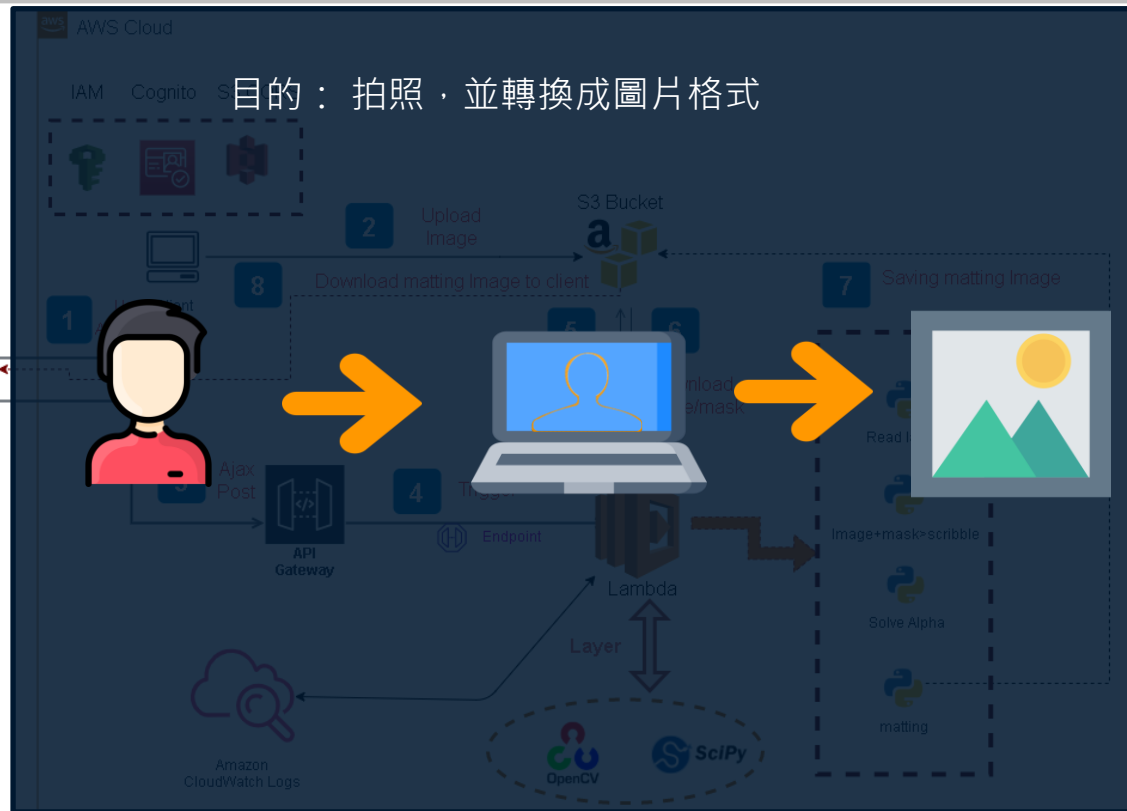
步驟5&6

步驟7

步驟8



Taking picture Application



## 流程2 上傳圖片至S3

Frontend

步驟1&2

步驟3&4

步驟5&6

步驟7

步驟8

Frontend



User

Channels

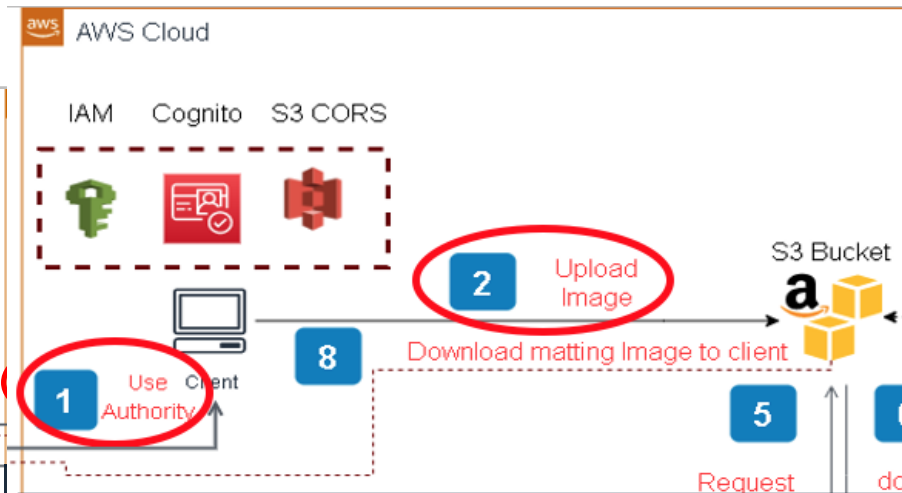


Mobile



Web

Taking picture Application



目的：

開權限給使用者，讓使用者可以上傳圖片至資料庫

流程：

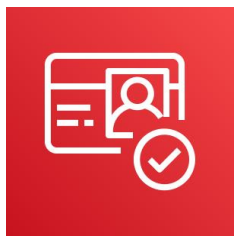
取得權限 → 產生token → 透過aws sdk上傳圖片



# 上傳S3的流程

在上傳S3之前，要先做以下事項

- 創好給user用的cognito identity pool
- 設定好S3的權限（隱私、CORS）



Amazon Cognito



Amazon S3



# 流程3&4 觸發無伺服器服務

Frontend ▼

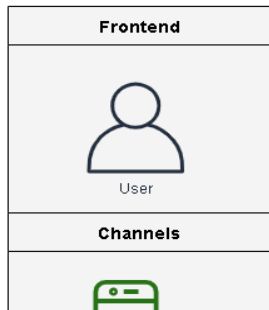
步驟1&2

步驟3&4 ✓

步驟5&6

步驟7

步驟8



目的：

設定一個可以讓外部連到的api、並觸發lambda函式

流程：

上傳結束 → □ 觸發api 端口  
→ □ 觸發lambda function



# 遭遇問題

- Lambda function trigger的問題
  - 解決方法：取消用s3作為lambda function的trigger，改採用API的方式
- API gateway timeout：超過可以容許的回應時間
  - 解決辦法：了解自己演算法的bottle neck，並做改善
- API 的部分雖可以用curl去測試，但在debug時資訊量不夠
  - 解決方法：可以透過cloudwatch去看log





# 流程5&6 連結S3及運算

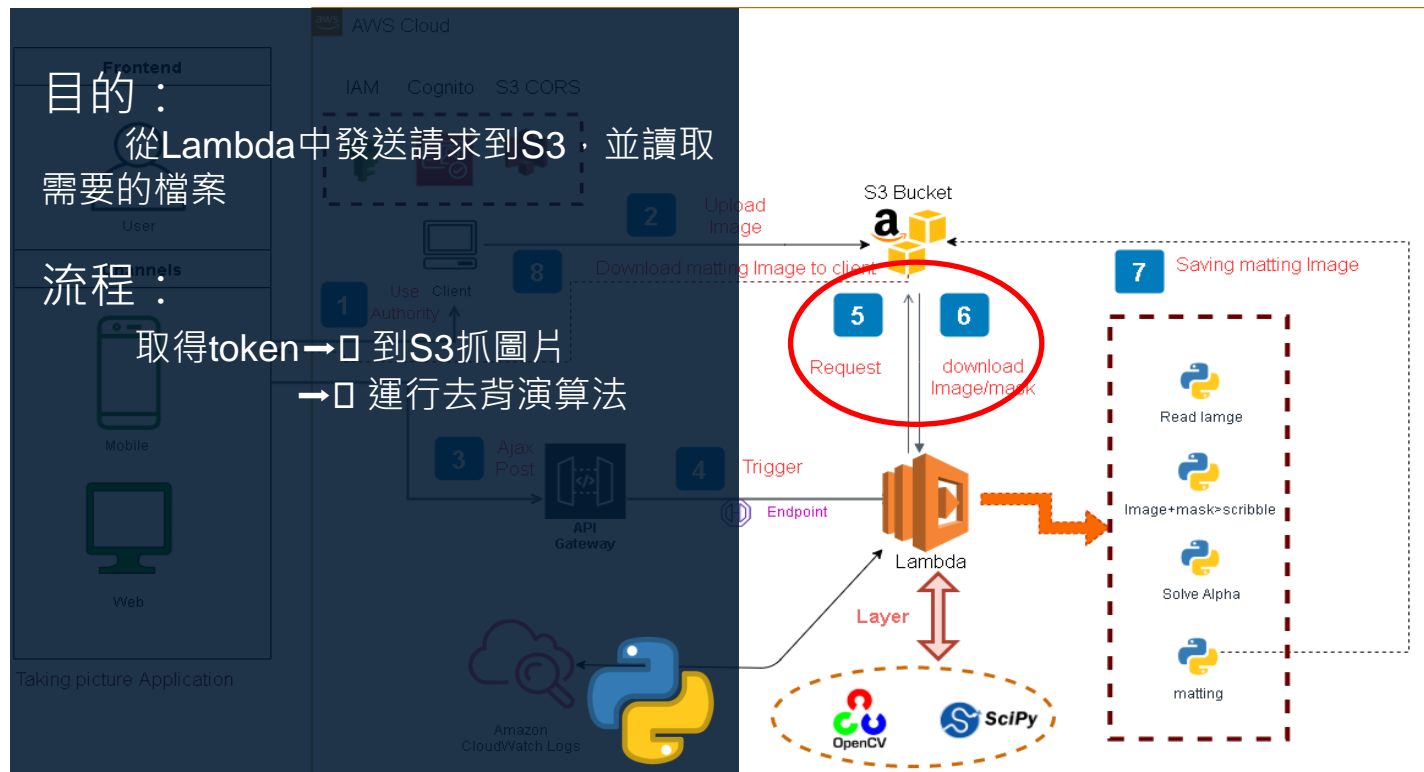
Frontend	▼
步驟1&2	
步驟3&4	
步驟5&6	✓
步驟7	
步驟8	

目的：

從Lambda中發送請求到S3，並讀取需要的檔案

流程：

取得token → 到S3抓圖片  
→ 運行去背演算法



# 流程7 儲存運算結果

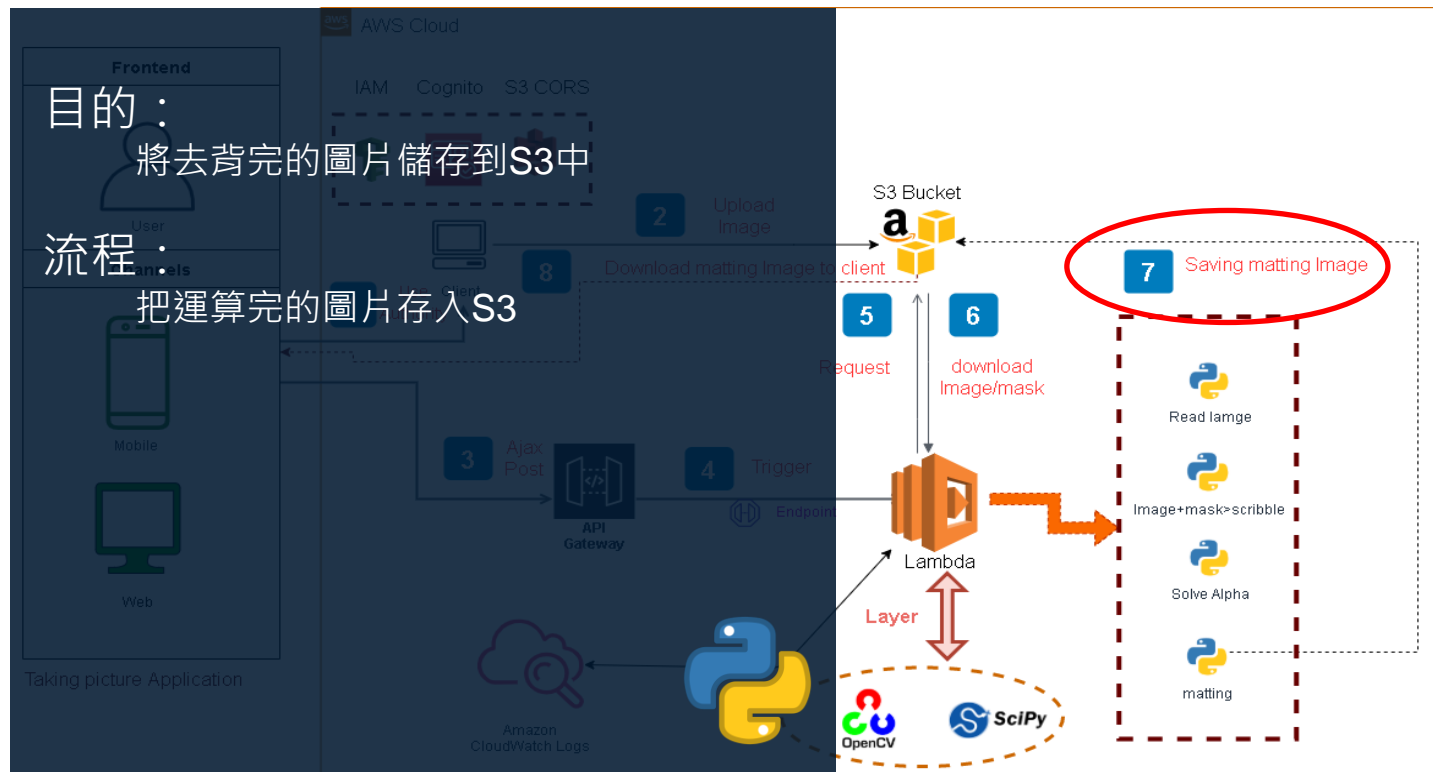
Frontend	▼
步驟1&2	
步驟3&4	
步驟5&6	
步驟7	✓
步驟8	

目的：

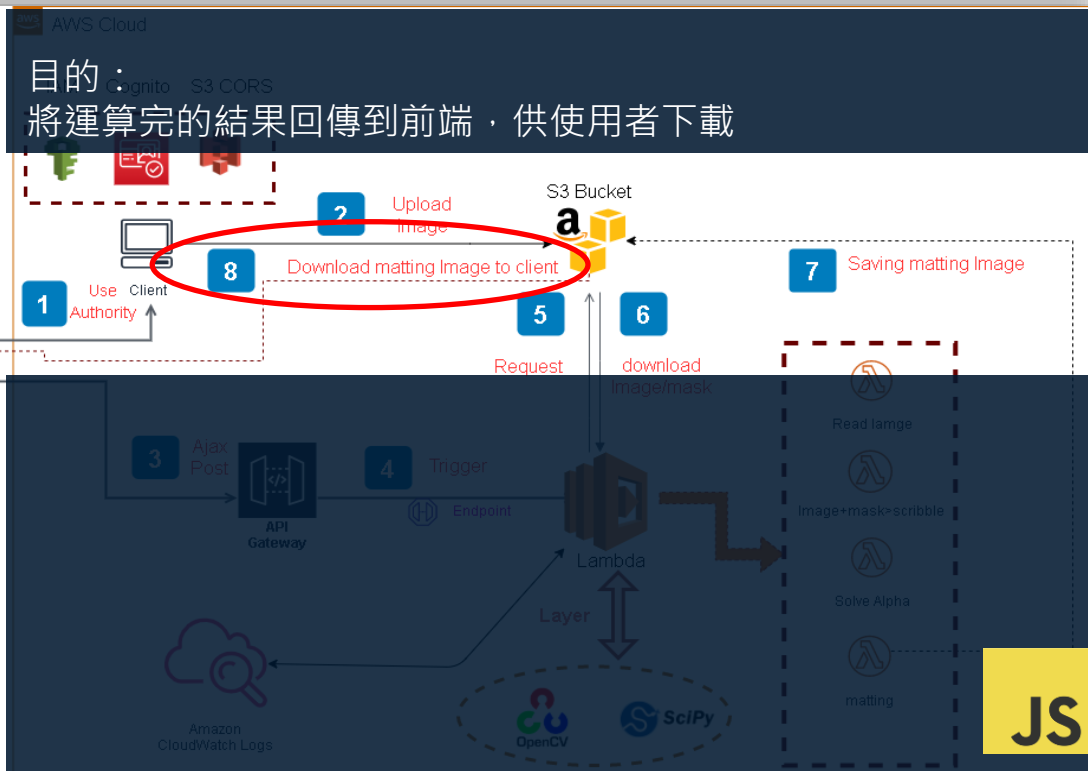
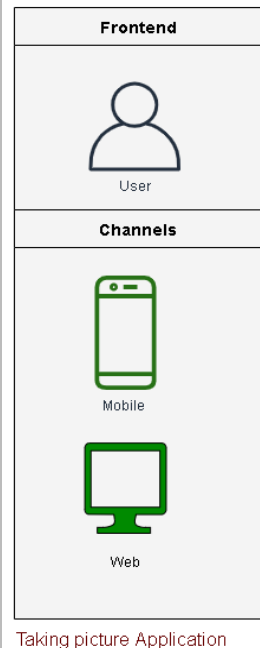
將去背完的圖片儲存到S3中

流程：

把運算完的圖片存入S3



# 流程8 返回結果至前端



# 遇到問題



## 版控

就算可以直接在Lambda上改，  
但要記得copy一份到地端

## 開發時間

目標明確、善用資源



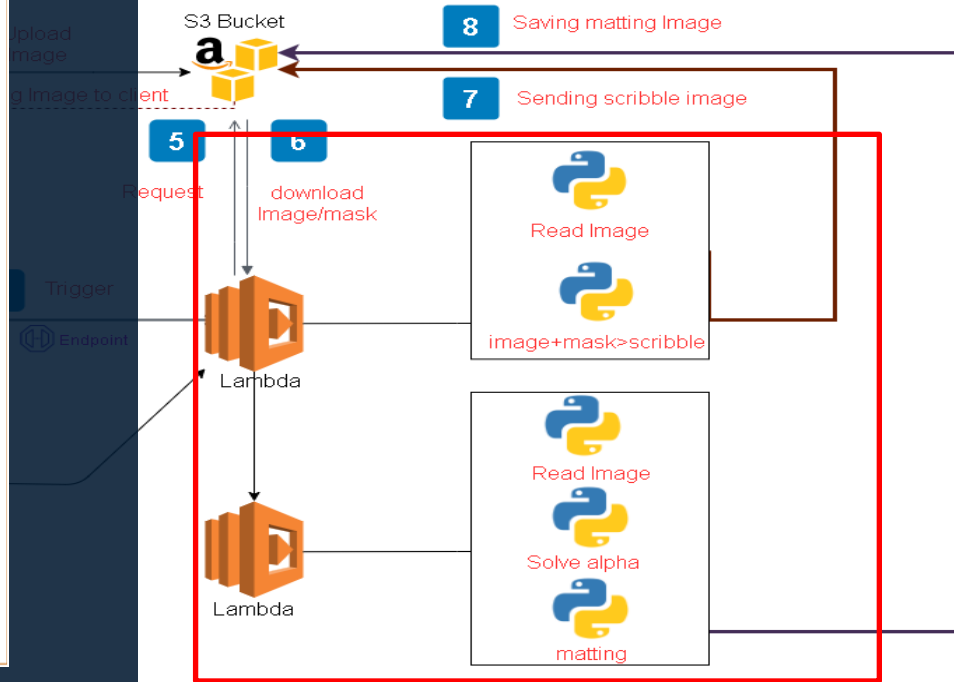
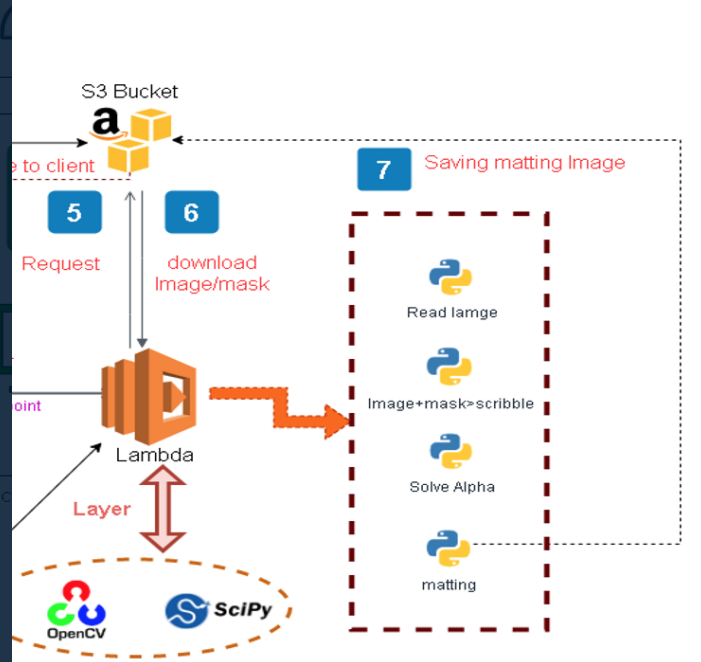


Demo time



## 軟體架構圖 2.0

### 優化版本，降低系統耦合度



# Future Improvement

## ■ Front end

增加去背效果

## ■ Algorithm

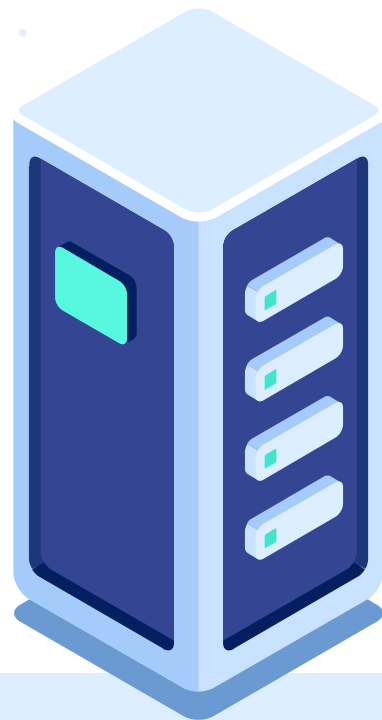
加快服務效率

## ■ ML/DL

#WhatifWeCould



4



實習心得

What if we could



**THANK  
YOU**

