

# AWS 기반 애플리케이션

2022. 4. 23

정 준 수 PhD

# Cloud Computing 환경이란?

## Hardware의 추상화

※ 전력과 네트워크 Failure에 대한 비상계획(Contingency Plan) 이 요구됨

# 이번 과정에서 다룰 내용들

- **Severless 환경**

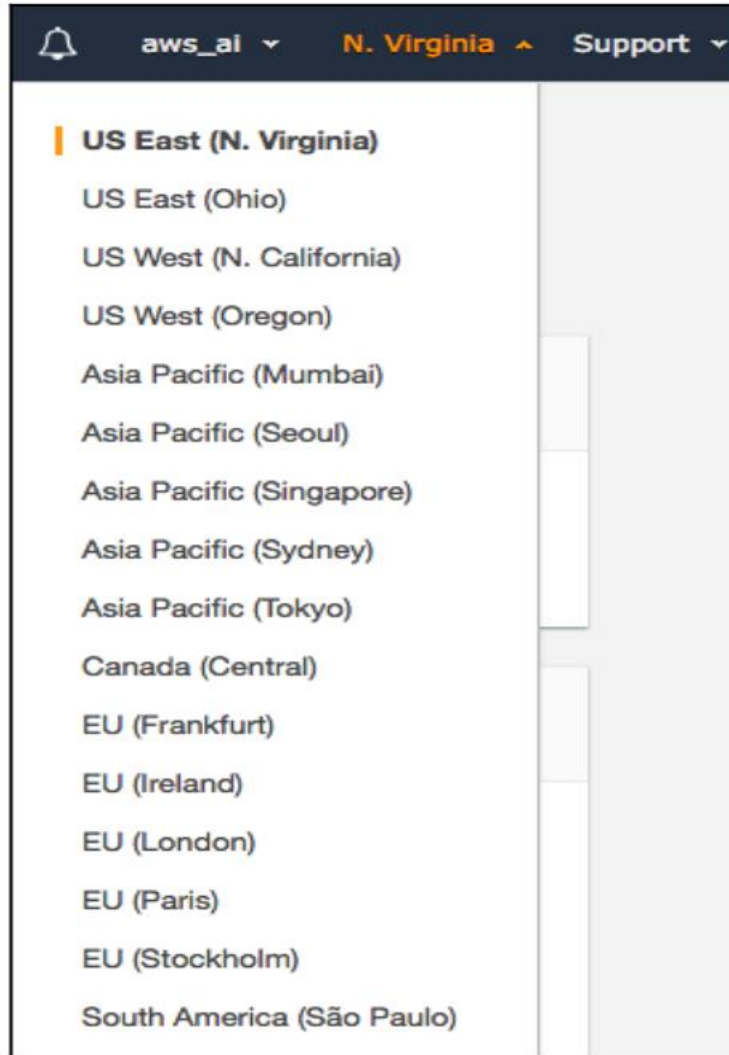
**EC2, S3**

**Lambda 함수, CloudWatch**

**APIGateway**

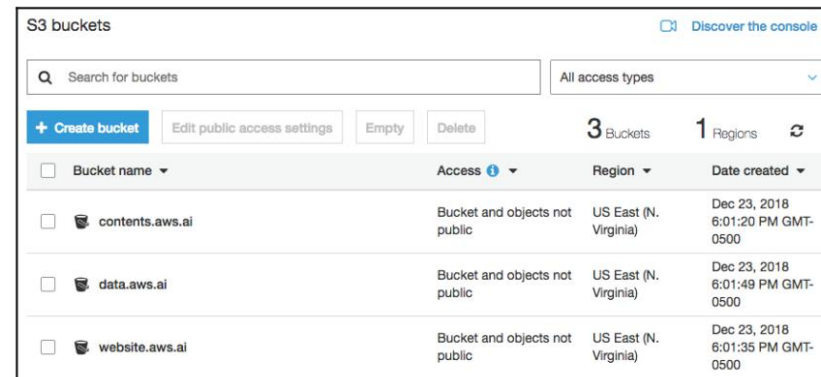
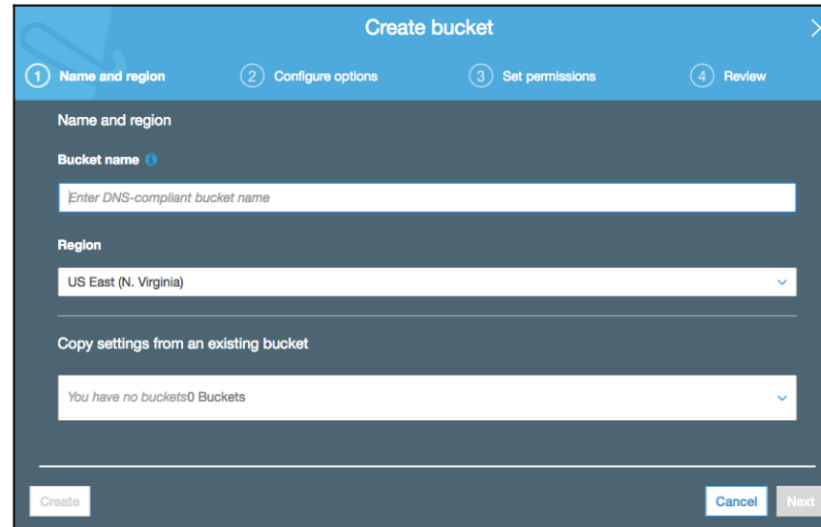
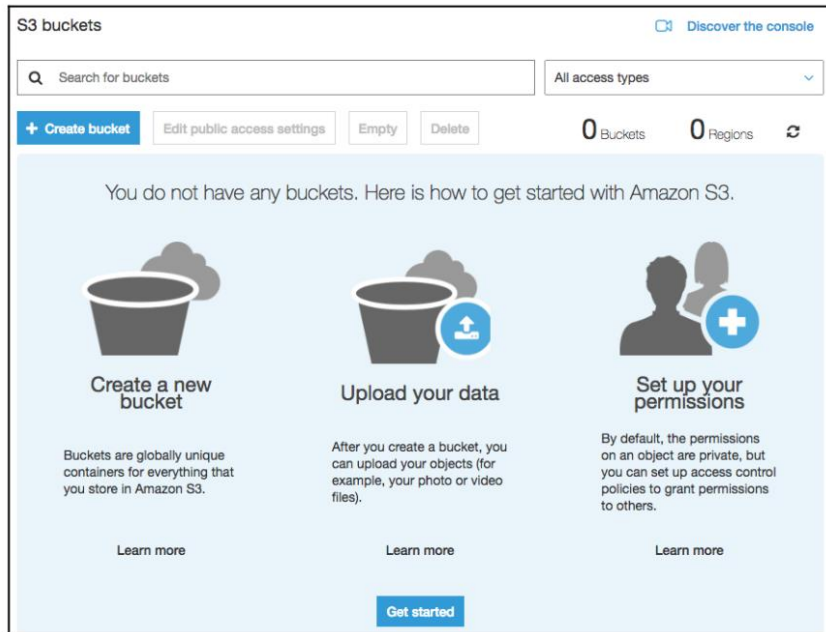
**Chalice framework**

# AWS 서비스 Region 선택하기



각 지역에 클라우드 데이터 센터를 구축하고 있는데 Region별로 AWS 서비스 배포 현황이 다르다. 따라서, 모든 Region에서 전체 AWS 서비스를 사용할 수 있는 것은 아니다.

# AWS – S3 사용하기(버킷 만들기)



- 1.버킷 만들기 클릭
- 2.버킷 이름, 리전 입력
- 3.퍼블릭 액세스 설정
- 4.버킷버전관리, 기본 암호화
- 5.버킷 생성 확인
- 6.업로드 시도
- 7.업로드된 사진 확인
- 8.업로드 이미지 확인하기

# AWS Access Key Pair 생성

The screenshot shows the AWS IAM Management Console for the user 'me'. The left sidebar contains navigation links for '대시보드', '액세스 관리', '사용자 그룹', '사용자', '역할', '정책', '자격 증명 공급자', '계정 설정', '보고서 액세스', '액세스 분석기', '아카이브 규칙', '분석기', '설정', '자격 증명 보고서', '조직 활동', and 'SCP(서비스 제어 정책)'. The main content area is titled '요약' and shows user details: '사용자 ARN: arn:aws:iam::213480204295:user/me', '경로: /', and '생성 시간: 2022-02-05 19:58 UTC+0900'. Below this, there are tabs for '권한', '그룹', '태그', '보안 자격 증명', and '액세스 관리자'. The '보안 자격 증명' tab is active, showing '로그인 자격 증명' with a console login link. The '액세스 키' section explains that access keys are used for AWS CLI, PowerShell, AWS SDK, and direct AWS API calls. It includes a warning: '비밀 키는 생성 시에만 보거나 다운로드할 수 있습니다. 기존 비밀 키를 잘못 보관한 경우 새 액세스 키를 생성하십시오. 자세히 알아보기'. A red circle highlights the '액세스 키 만들기' button. Below this is a table of existing access keys:

액세스 키 ID	생성 완료	마지막 사용	상태
AKIATDNDJNAD6HFRT3WC	2022-02-05 19:58 UTC+0900	ap-northeast-1에서 2022-0...	(활성)   비활성화

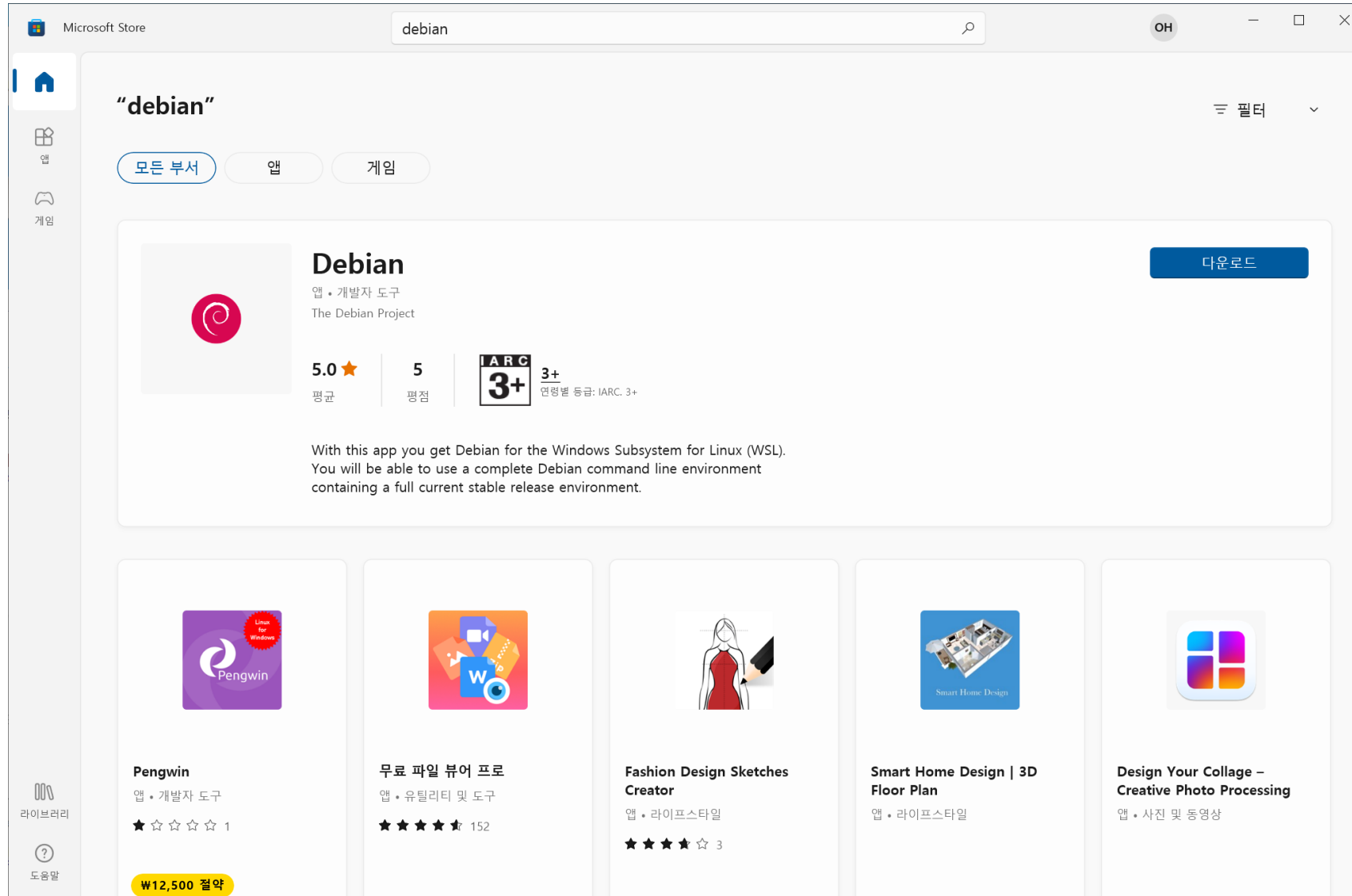
At the bottom, there is a section for 'AWS CodeCommit에 대한 SSH 키' with a link to '자세히 알아보기' and a button 'SSH 퍼블릭 키 업로드'.

# PC(Local) 환경구축: WSL

윈도우 환경에서 원하는 Linux를 실행

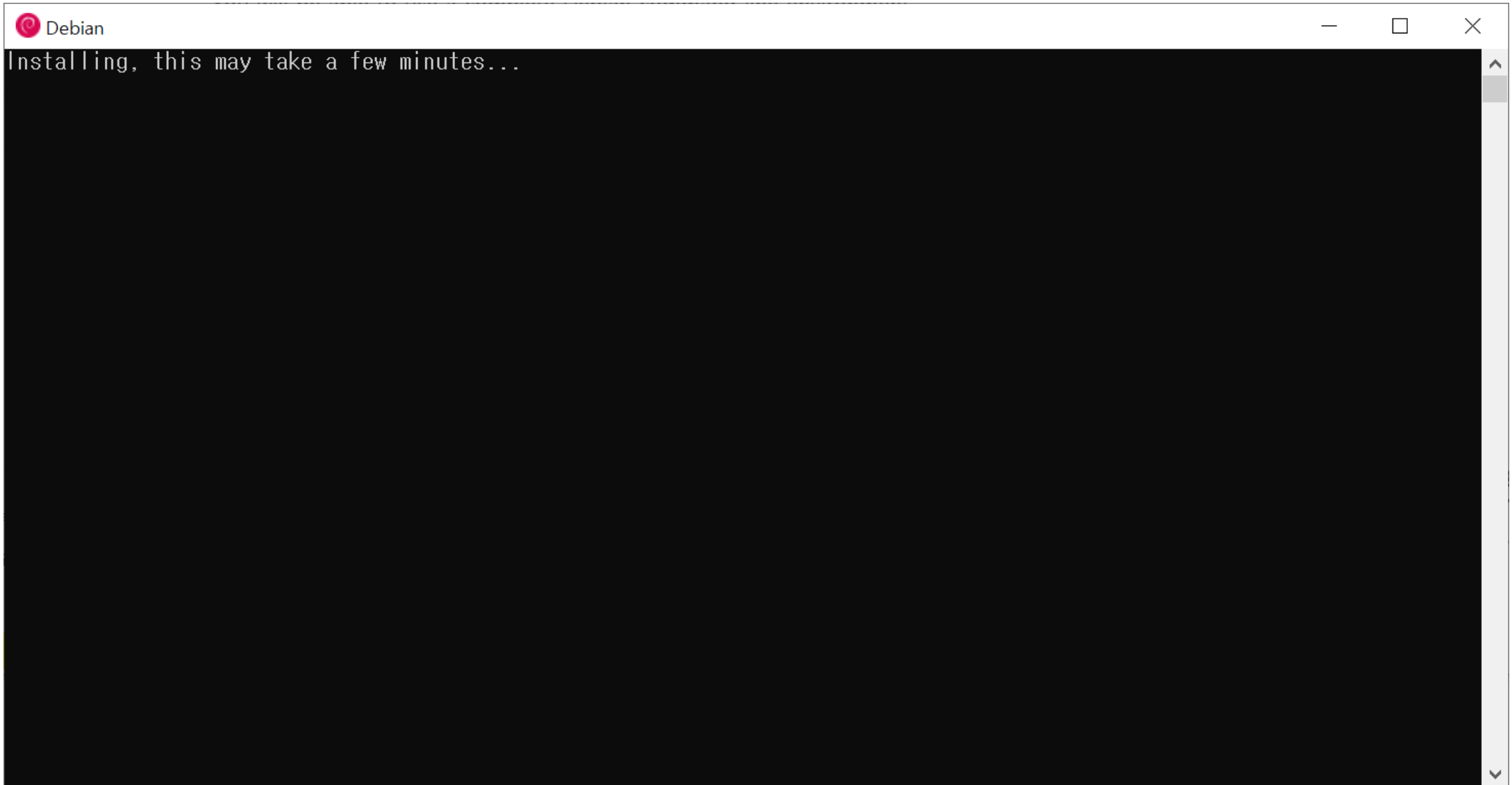
1. 관리자 권한으로 명령 프롬프트(CMD) 실행
2. <https://docs.microsoft.com/ko-kr/windows/wsl/tutorials/gui-apps>
3. C> Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-windows-Subsystem-Linux
4. 또는 <https://ivyit.tistory.com/264>
5. Microsoft Store 열기
6. Ubuntu 또는 Debian 선택

# Microsoft Store에서: Linux 선택





# Debian 설치 과정



# Debian 설치 과정

```
oakyo@DESKTOP-HAAIOJO: ~  
Installing, this may take a few minutes...  
Please create a default UNIX user account. The username does not need to match your Windows username.  
For more information visit: https://aka.ms/wslusers  
Enter new UNIX username: oakyo  
New password:  
Retype new password:  
passwd: password updated successfully  
Installation successful!  
oakyo@DESKTOP-HAAIOJO:~$
```

# Python 설치 과정

```
$ sudo apt update
```

```
$ sudo apt upgrade
```

```
$ sudo apt-get install wget
```

**Anaconda 설치 :** <https://repo.anaconda.com/archive/>

```
$ wget https://repo.anaconda.com/archive/Anaconda3-2021.11-Linux-x86\_64.sh
```

```
$ bash Anaconda3-2021.11-Linux-x86_64.sh
```

## PATH 추가

```
$ export PATH=/home/{login ID}/anaconda3/bin:$PATH
```

## Python version 확인

\$ python -version

## PC(Local) 파일 확인

\$ explorer.exe .

# AWS CLI 설치하기

```
$ sudo pip install awscli
```

```
$ aws --version
```

```
$ aws configure
```

설치 확인 방법

```
$ cd
```

```
$ ls -al
```

```
$ aws s3 ls
```

# AWS CLI 설치 확인

```
$ aws s3 ls s3://kosa-aws
```

```
$ aws rekognition detect-labels --image  
S3Object=W{Bucket=kosa-aws,Name=traffic-signal-  
1.jpgW}
```

```
KOSA_AWS_AI/CLI/
```

```
$ python object_detection_demo.py
```

KOSA\_AWS\_AI/CLI/

```
$ pip install boto3
```

```
$ python object_detection_demo.py
```

# Chalice Static Web Hosting

The screenshot shows the AWS S3 console interface for configuring static website hosting on a bucket named 'kosa-aws'. The left sidebar contains navigation options like '버킷' (Buckets), 'Storage Lens', and 'S3용 AWS Marketplace'. The main content area is titled '정적 웹 사이트 호스팅 편집' (Edit Static Website Hosting) and includes the following sections:

- 정적 웹 사이트 호스팅**: A note stating that the bucket is used for hosting websites and a link to '자세히 알아보기' (Learn more).
- 정적 웹 사이트 호스팅**: Radio buttons for '비활성화' (Deactivated) and '활성화' (Activated), with '활성화' selected.
- 호스팅 유형**: Radio buttons for '정적 웹 사이트 호스팅' (Static Website Hosting) and '객체에 대한 요청 리디렉션' (Request redirection for objects). '정적 웹 사이트 호스팅' is selected, with a note that bucket endpoints are used as website URLs and a link to '자세히 알아보기' (Learn more).
- 고객이 웹 사이트 엔드포인트의 콘텐츠에 액세스할 수 있게 하려면 모든 콘텐츠를 공개적으로 읽기 가능하도록 설정해야 합니다.**: A warning box stating that to allow public access to content, the bucket's S3 public access settings must be edited. It provides a link to 'Amazon S3 퍼블릭 액세스 차단 사용' (Use Amazon S3 public access blocking) for more information.
- 인덱스 문서**: A text input field for the index document, containing 'index.html'.
- 오류 문서 - 선택 사항**: A text input field for the error document, containing 'error.html'.
- 리디렉션 규칙 - 선택 사항**: A note stating that JSON-based redirect rules are automatically created for specific content and a link to '자세히 알아보기' (Learn more).

The bottom of the console shows a table with one row, indicating the configuration is complete. The footer of the console displays '© 2022, Amazon Web Services, Inc. 또는 계열사. 개인 정보 보호 약관 쿠키 기본 설정'.



# Chalice Static Web Hosting

The screenshot shows the Amazon S3 console interface for configuring static website hosting on a bucket named 'static-web-me'. The breadcrumb navigation is 'Amazon S3 > static-web-me > 정적 웹 사이트 호스팅 편집'. The main title is '정적 웹 사이트 호스팅 편집' with an 'Info' link.

**정적 웹 사이트 호스팅**  
이 버킷을 사용하여 웹 사이트를 호스팅하거나 요청을 리디렉션합니다. [자세히 알아보기](#)

**정적 웹 사이트 호스팅**

- ☐ 비활성화
- ☒ 활성화

**호스팅 유형**

- ☒ 정적 웹 사이트 호스팅  
버킷 엔드포인트를 웹 주소로 사용합니다. [자세히 알아보기](#)
- ☐ 객체에 대한 요청 리디렉션  
요청을 다른 버킷 또는 도메인으로 리디렉션합니다. [자세히 알아보기](#)

**고객이 웹 사이트 엔드포인트의 콘텐츠에 액세스할 수 있게 하려면 모든 콘텐츠를 공개적으로 읽기 가능하도록 설정해야 합니다. 이렇게 하려면, 버킷에 대한 S3 퍼블릭 액세스 차단 설정을 편집하면 됩니다. 자세한 내용은 Amazon S3 퍼블릭 액세스 차단 사용 참조하십시오.**

**인덱스 문서**  
웹 사이트의 홈 페이지 또는 기본 페이지를 지정합니다.

index.html

**오류 문서 - 선택 사항**  
오류가 발생하면 반환됩니다.

error.html

**리디렉션 규칙 - 선택 사항**  
JSON으로 작성된 리디렉션 규칙은 특정 콘텐츠에 대한 웹 페이지 요청을 자동으로 리디렉션합니다. [자세히 알아보기](#)

1

의견 한국어 ▼ © 2022, Amazon Web Services, Inc. 또는 계열사. 개인 정보 보호 약관 쿠키 기본 설정

# Bucket 소유자의 정식 ID 확인

The screenshot shows the AWS S3 console interface. The left sidebar contains navigation links for '버킷' (Buckets), '액세스 지점' (Access Points), '객체 Lambda 액세스 지점' (Object Lambda Access Points), '다중 리전 액세스 지점' (Multi-Region Access Points), '배치 작업' (Batch Jobs), 'S3용 액세스 분석기' (S3 Access Analyzer), '이 계정의 퍼블릭 액세스 차단 설정' (Public Access Block Settings), 'Storage Lens' (with sub-links '대시보드' and 'AWS Organizations 설정'), '기능 스포트라이트' (3 items), and 'S3용 AWS Marketplace'.

The main content area displays two informational messages at the top:

- 이 버킷에 대해 퍼블릭 액세스 차단 설정이 활성화되어 있기 때문에 퍼블릭 액세스가 차단됩니다. 활성화된 설정을 확인하려면 이 버킷의 퍼블릭 액세스 차단 설정을 확인하세요. [Amazon S3 퍼블릭 액세스 차단 사용](#)에 대해 자세히 알아보기
- 중복 피부여자에 대한 통합 액세스 권한 부여가 콘솔에 표시됩니다. 전체 ACL 목록을 보려면 Amazon S3 REST API, AWS CLI 또는 AWS SDK를 사용하세요.

Below the messages is a table titled '퍼블릭 액세스' (Public Access) showing bucket ACLs:

퍼블릭 액세스	객체	버킷 ACL
<b>정식 ID 복사됨</b> 정식 ID: 963bb1530939f6612d66a80617f7bf65ecfe113275038ca6a52eefbca0658053	나열, 쓰기	읽기, 쓰기
모든 사람(퍼블릭 액세스) 그룹: <a href="http://acs.amazonaws.com/groups/global/AllUsers">http://acs.amazonaws.com/groups/global/AllUsers</a>	-	-
인증된 사용자 그룹(AWS 계정이 있는 모든 사용자) 그룹: <a href="http://acs.amazonaws.com/groups/global/AuthenticatedUsers">http://acs.amazonaws.com/groups/global/AuthenticatedUsers</a>	-	-
S3 로그 전달 그룹 그룹: <a href="http://acs.amazonaws.com/groups/s3/LogDelivery">http://acs.amazonaws.com/groups/s3/LogDelivery</a>	-	-

Below the table is a section for 'CORS(Cross-origin 리소스 공유)' with a description and a 'JSON으로 작성된 CORS 구성은 한 도메인에 로드되어 다른 도메인의 리소스와 상호 작용하는 클라이언트 웹 애플리케이션에 대한 방법을 정의합니다. [자세히 알아보기](#)' link. There is a '편집' (Edit) button.

At the bottom, there is a section for '표시할 구성 없음' (No configurations to display) with a '복사' (Copy) button.

The footer of the console shows '의견' (Feedback), '한국어' (Korean), '© 2022, Amazon Web Services, Inc. 또는 계열사.', '개인 정보 보호' (Privacy), '약관' (Terms), and '쿠키 기본 설정' (Cookie Settings).

# ACL(엑세스 제어 목록 확인)

Googlestatic-web-me - S3 bucket객체 탐지기403 Forbidden

s3.console.aws.amazon.com/s3/bucket/static-web-me/property/acl/edit?region=ap-northeast-1

aws서비스서비스, 기능, 블로그, 설명서 등을 검색합니다.[Alt+S]글로벌me @ 2134-8020-4295

ACL(엑세스 제어 목록)  
다른 AWS 계정에 기본 읽기/쓰기 권한을 부여합니다. 자세히 알아보기

피부여자	객체	버킷 ACL
버킷 소유자(AWS 계정) 정식 ID: 963bb1530939f6612d66a80617f7bf65ecfe113275038ca6a52eefbca0658053	<input checked="" type="checkbox"/> 나열 <input checked="" type="checkbox"/> 쓰기	<input checked="" type="checkbox"/> 읽기 <input checked="" type="checkbox"/> 쓰기
모든 사람(퍼블릭 액세스) 그룹: http://acs.amazonaws.com/groups/global/AllUsers	<input type="checkbox"/> 나열 <input type="checkbox"/> 쓰기	<input checked="" type="checkbox"/> 읽기 <input type="checkbox"/> 쓰기
인증된 사용자 그룹(AWS 계정이 있는 모든 사용자) 그룹: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	<input type="checkbox"/> 나열 <input type="checkbox"/> 쓰기	<input type="checkbox"/> 읽기 <input type="checkbox"/> 쓰기
S3 로그 전달 그룹 그룹: http://acs.amazonaws.com/groups/s3/LogDelivery	<input type="checkbox"/> 나열 <input type="checkbox"/> 쓰기	<input type="checkbox"/> 읽기 <input type="checkbox"/> 쓰기

모든 사용자 또는 인증된 사용자 그룹 피부여자에게 액세스 권한을 부여하면 누구나 이 버킷의 객체에 액세스할 수 있습니다.

자세히 알아보기

☒ 이러한 변경 사항이 객체와 버킷에 미치는 영향을 이해합니다.

다른 AWS 계정에 대한 액세스  
리소스와 연결된 다른 AWS 계정이 없습니다.

피부여자 추가

취소변경 사항 저장

의견한국어© 2022, Amazon Web Services, Inc. 또는 계열사.개인 정보 보호약관쿠키 기본 설정

# AWS 서버리스(Serverless)

서버리스([Serverless: The Serverless Application Framework](#))는 AWS 람다를 비롯해 애저 Function, 구글 클라우드 Function 등을 지원하는 범용 서버리스 프레임워크입니다.

AWS 람다의 경우, 람다 함수 배포 뿐 아니라 클라우드 포메이션 스택을 통해 **IAM, DynamoDB, S3, 클라우드 와치 이벤트(CloudWatch Events)** 등 아마존 리소스를 함께 지원하고 있습니다.

# AWS Lambda

AWS 람다AWS Lambda는 [아마존 웹 서비스](#)Amazon Web Services에서 제공하는 서버리스 컴퓨팅 서비스입니다.

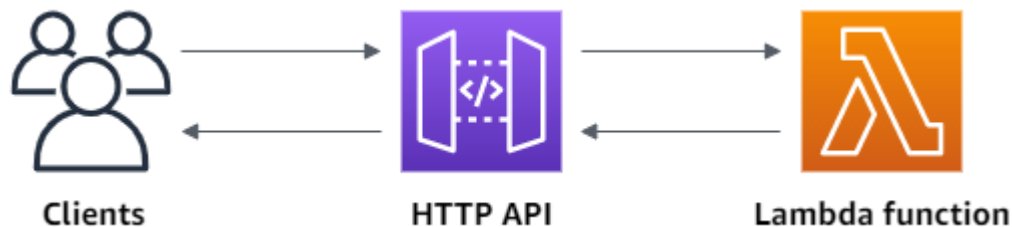
2014년 11월 AWS 리인벤트AWS re:Invent에서 처음 발표 되었습니다. 서버리스 컴퓨팅은 애플리케이션을 실행하기 위한 별도의 서버 셋업 없이 곧바로 코드를 실행해주는 서비스를 의미하며, 고정 비용 없이 사용 시간에 대해서만 비용이 발생합니다. 아마존 EC2Amazon EC2는 현재 초 단위로 비용을 계산하는 반면\* 람다는 1ms당 요금을 계산해 정확히 사용한만큼만 비용이 발생합니다.

# API Gateway

이번 시작하기 연습에서는 서버리스 API를 생성합니다. 서버리스 API를 사용하면 서버를 프로비저닝하고 관리하는 데 시간을 소비하는 대신 애플리케이션에 집중할 수 있습니다. 이 연습은 완료하는 데 20분을 넘지 않으며 [AWS 프리 티어](#) 내에서 가능합니다.

먼저, AWS Lambda 콘솔을 사용하여 Lambda 함수를 생성합니다. 그런 다음 API Gateway 콘솔을 사용하여 HTTP API를 생성합니다. 그런 다음 API를 호출합니다.

HTTP API를 호출하면 API Gateway는 요청을 Lambda 함수로 라우팅합니다. Lambda는 Lambda 함수를 실행하고 API Gateway에 응답을 반환합니다. 그리고 나면 API Gateway가 응답을 반환합니다.



# Chalice Python Serverless Microframework

<https://github.com/aws/chalice>

```
$ pip install chalice
```

```
$ sudo apt-get install curl
```

```
$ sudo apt-get install httpie
```

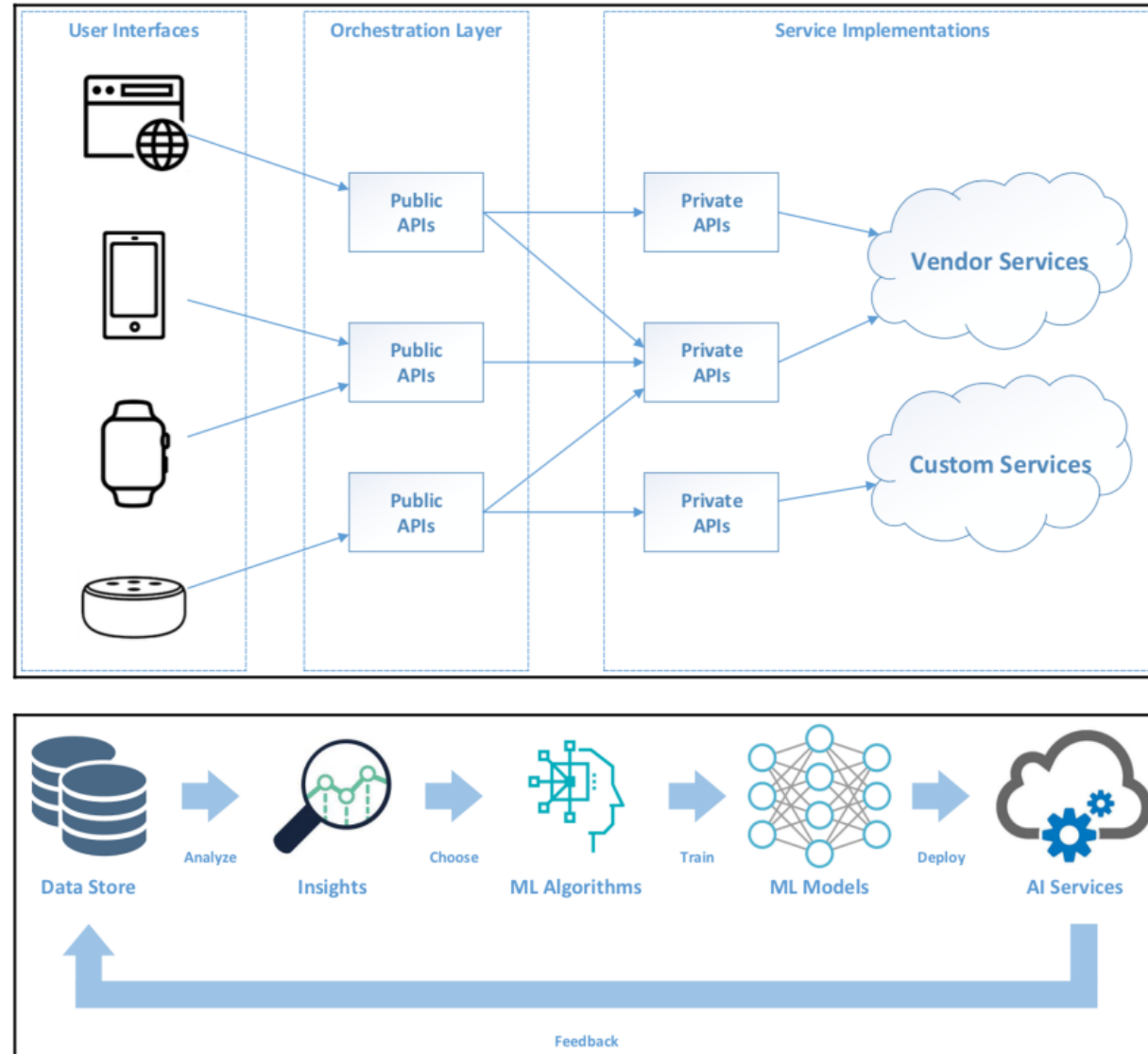
```
$ chalice new-project helloworld
```

```
$ chalice local
```

```
$ chalice deploy
```

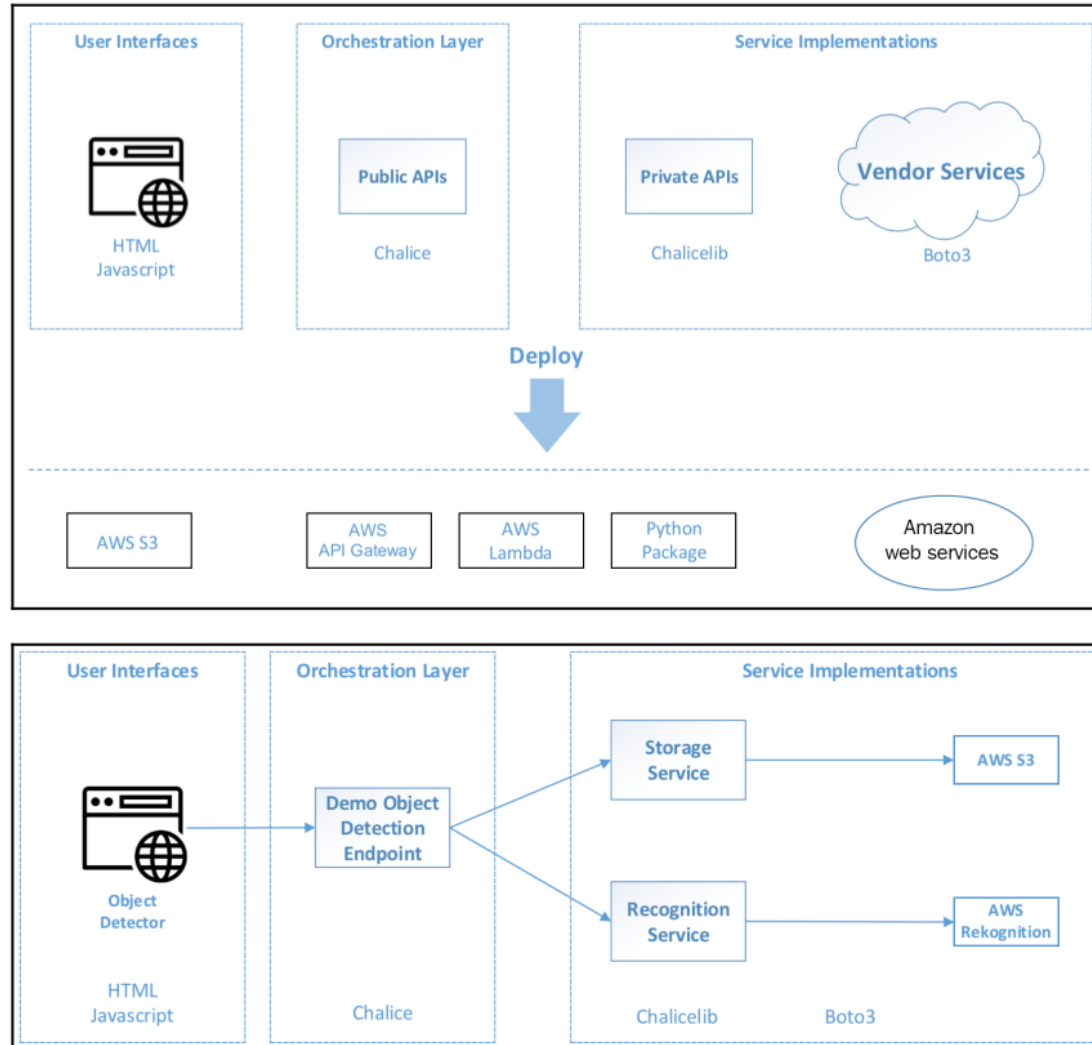
**Local에서 AI app개발 환경을 제공**

# AWS AI 애플리케이션의 아키텍처 이해





# Chalice: AI 애플리케이션 아키텍처



# Project 1: AWS Rekognition (Local)

**[https://github.com/JSJeong-me/KOSA\\_AWS\\_AI/Project01](https://github.com/JSJeong-me/KOSA_AWS_AI/Project01)**

```
$ cd Project01
```

```
$ chalice new-project Capabilities
```

```
$ cd Capabilities
```

```
$ mkdir chalicelib
```

```
$ chalice local
```

```
$ curl http://127.0.0.1:8000/demo-object-detection
```

# Project02: AWS Rekognition (Web UI)

**[https://github.com/JSJeong-me/KOSA\\_AWS\\_AI/Project02](https://github.com/JSJeong-me/KOSA_AWS_AI/Project02)**

\$ chalice local

./Website/index.html 실행

# Project03: AWS Rekognition (AWS Deploy)

**[https://github.com/JSJeong-me/KOSA\\_AWS\\_AI/Project03](https://github.com/JSJeong-me/KOSA_AWS_AI/Project03)**

```
$ cd Project03
```

```
$ cd Capabilities
```

```
$ chalice deploy
```

```
$ curl https://6pku28v9ok.execute-api.ap-northeast-1.amazonaws.com/api/demo-object-detection
```

# AWS Rekognition

Amazon Rekognition

Metrics

Demos

**Object and scene detection**

Image moderation

Facial analysis

Celebrity recognition

Face comparison

Text in image

Video Demos

Video analysis

Additional Resources

Getting started guide

Download SDKs

Developer resources


Pricing

FAQ

Forum

## Object and scene detection

Rekognition automatically labels objects, concepts and scenes in your images, and provides a confidence score.



Done with the demo?  
[Learn more](#)

▼ Results



Transportation	98.8 %
Vehicle	98.8 %
Automobile	98.8 %
Car	98.8 %
Person	98.3 %
Human	98.3 %

Show more

► Request

► Response

Choose a sample image

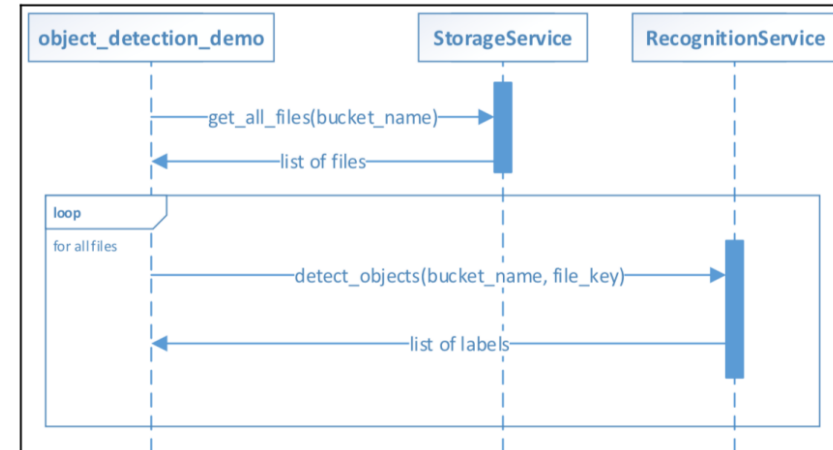


Use your own image

Image must be .jpg or .png format and no larger than 5MB. Your image isn't stored.

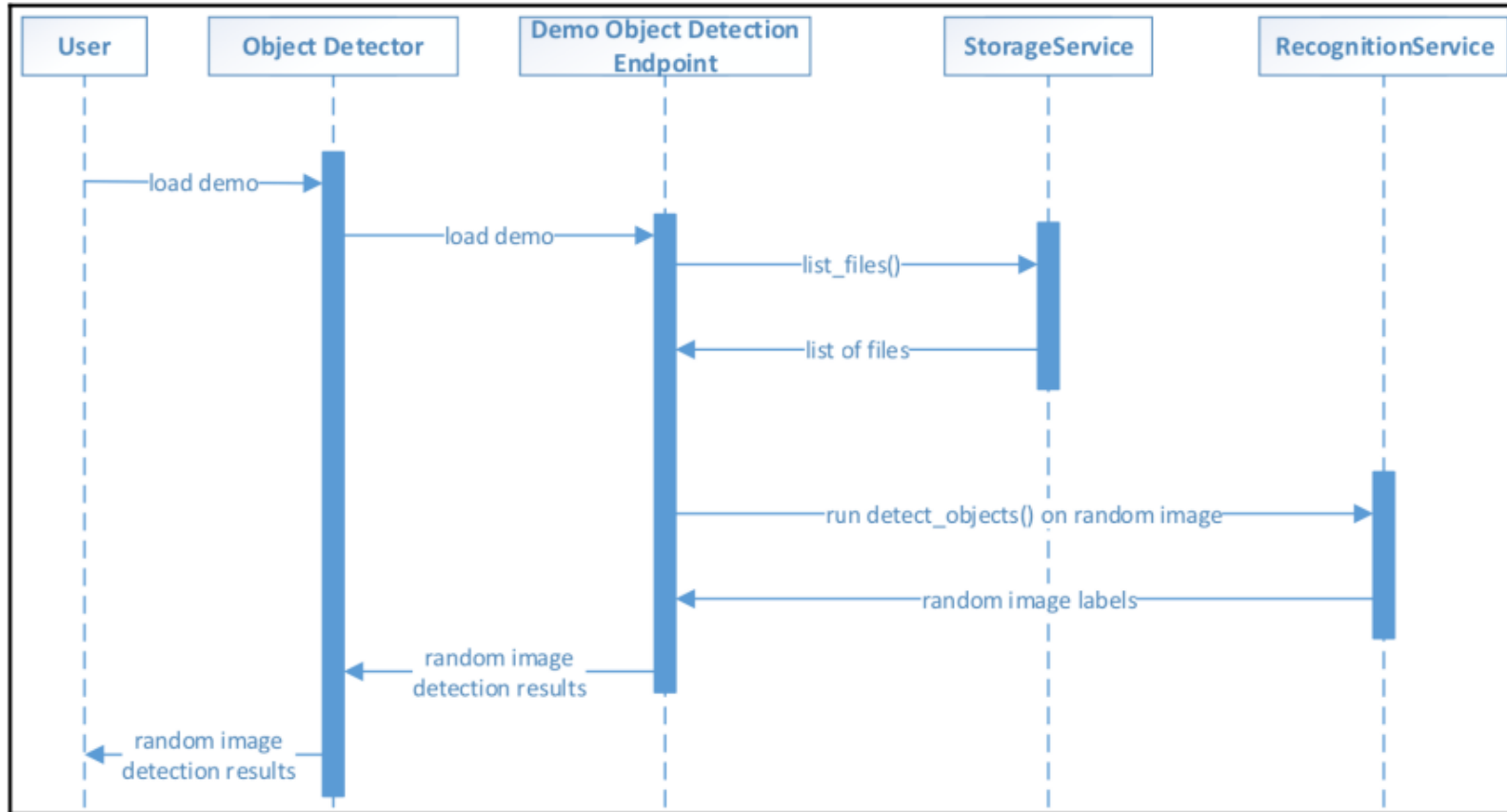
[Upload](#) or drag and drop

Use image URL [Go](#)



<https://ap-northeast-1.console.aws.amazon.com/rekognition/home?region=ap-northeast-1#/label-detection>

# Rekognition Prj. Component간의 상호작용



정 준 수 / Ph.D ( jsjeong@hansung.ac.kr )

- 前) 삼성전자 연구원
- 前) 삼성의료원 (삼성생명과학연구소)
- 前) 삼성SDS (정보기술연구소)
- 現) (사)한국인공지능협회, AI, 머신러닝 강의
- 現) 한국소프트웨어산업협회, AI, 머신러닝 강의
- 現) 서울디지털재단, AI 자문위원
- 現) 한성대학교 교수(겸)
- 전문분야: Computer Vision, 머신러닝(ML), RPA
- <https://github.com/JSJeong-me/>

