**디지털 핵심 실무인재 양성사업(K-Digital Training)**

**3조 1차 프로토타입**

**2023년 01월 12 일**

1차 구성(1/11~1/15)

| 시스템 구성도 |  | | |
| --- | --- | --- | --- |
|
|
|
|
|
|
|
|
|
|
|
|
| comment | 주제: Web Server를 S3로 생성해서 이미지로 저장 기능  1.구성:  -Client가 S3(Web-server)로 Image 데이터 전송(Api gateway-Lambda로 presigned url 생성후 url로 image s3 upload)  -Model Server와 마운트 된 S3로부터 image 데이터 받고 RDS에 결과 저장  -Lambda로부터 결과 get으로 호출해 front에 도출  2.문제점:  - S3 이미지 저장후 모델로 전송하는 이벤트 발생 방법 애매  - CORS허용 설정을 해도 ajax통신시 SOP에 걸림  -원인을 알 수 없는 403 Error  - 처음 해보는 방식이라 Error 발생시 발견 힘듦  - Image데이터가 json이 아닌 Formdata이기 때문에 Lambda-Api 구성이 힘듦    3. 방안:  - 이벤트=EC2와 Object Mount방식으로 이벤트 전달 구상  -CORS = Moseif Origins & CORS Changer 사용  - 잘 만들어진 Django Framework 사용 | | |

2차 구성(1/16~1/19)

| 시스템 구성도 |  | | |
| --- | --- | --- | --- |
|
|
|
|
|
|
|
|
|
|
|
|
| comment | 주제 :Docker Container를 통한 Infra 구성  1. 절차:    -Client가 EC2 Server에 Image Data 전송  - EC2내 Port Forwarding된 NGINX Container에서 요청 받아 Django Server Container로 Proxy  - Django Server에서 받은 Image 데이터 Model Container로 재요청후 연산된 결과값 Response  - 받은 결과값 해석해 Render후 Client Browser에 전달  2.문제점:  - Key 및 보안 취약  - Versioning이 힘듦  - 서버 다운 및 다중 Traffic 취약  3. 방안 및 개선점  -Key = Docker secret 사용, HTTPS 사용  -CICD 툴 검토(Jenkins)  - EKS로 Architecture upgrade시켜 Autoscaling, Load Balancing 검토  - AWS Autoscaling Load Balancing 사용  - Front 개선  - Login기능 구현  - 개발,배포용 분리  -DNS 설정 | | |