

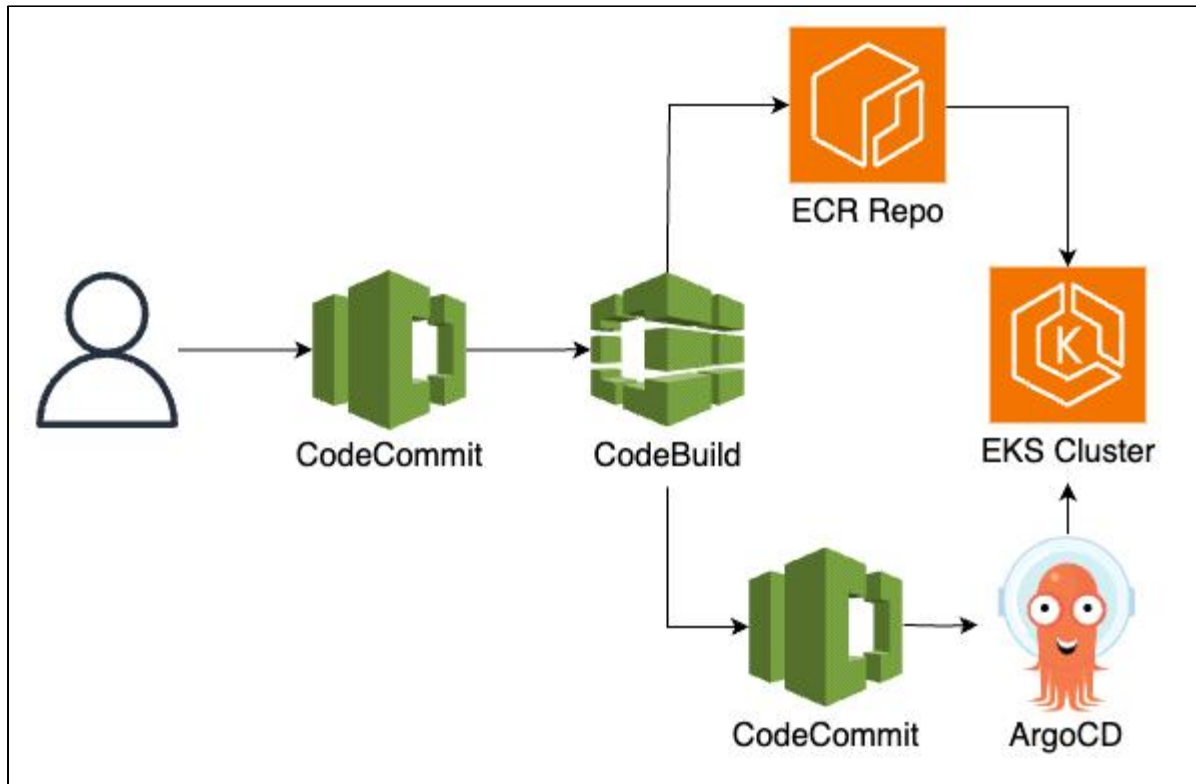
# 2024년도 전국기능경기대회

직 종 명	클라우드컴퓨팅	과 제 명	Small Challenge	과제번호	제 2과제
경기시간	4시간	비 번 호		심사위원 확 인	(인)

## 1. 요구사항

EKS와 CodeSeries를 사용하여 CI/CD 파이프라인을 구축해야 합니다. EKS상에서 구동되는 컨테이너 이미지가 빌드되어 ECR에 업로드되면 ArgoCD를 통해 업로드된 이미지로 컨테이너에 무중단 배포 되어야 합니다. 요구사항에 맞게 적절한 파이프라인을 구성해야 합니다.

### 다이어그램



## Software Stack

AWS	개발언어/프레임워크
- EKS	- Python
- CodeCommit	- ArgoCD
- CodeBuild	- Argo rollout
- CodePipeline	- Argo rollouts CLI
- ELB	- kubectl
- ECR	- AWS Load Balancer Controller

## 2. 선수 유의사항

- 1) 기계 및 공구 등의 사용 시 안전에 유의하시고, 필요 시 안전장비 및 복장 등을 착용하여 사고를 예방하여 주시기 바랍니다.
- 2) 작업 중 화상, 감전, 찰과상 등 안전사고 예방에 유의하시고, 공구나 작업도구 사용 시 안전보호구 착용 등 안전수칙을 준수하시기 바랍니다.
- 3) 작업 중 공구의 사용에 주의하고, 안전수칙을 준수하여 사고를 예방하여 주시기 바랍니다.
- 4) 경기 시작 전 가벼운 스트레칭 등으로 긴장을 풀어주시고, 작업도구의 사용 시 안전에 주의하십시오.
- 5) 선수의 계정에는 비용의 제한이 존재하며, 이보다 더 높게 요금이 부과될 시 계정 사용이 불가능할 수 있습니다.
- 6) 문제에 제시된 괄호 박스 ◇는 변수를 뜻함으로 선수가 적절히 변경하여 사용해야 합니다.
- 7) EC2 인스턴스의 TCP 80/443 outbound는 anyopen하여 사용할 수 있도록 합니다.
- 8) 과제의 Bastion 서버에서 대부분의 채점이 이루어짐으로 인스턴스를 생성하지 않았거나 종료된 상태면 채점이 불가능하니 각별히 주의하도록 합니다.
- 9) 과제 종료 시 진행 중인 테스트를 모두 종료하여 서버에 부하가 발생하지 않도록 합니다.
- 10) 별도 언급이 없는 경우, ap-northeast-2 리전에 리소스를 생성하도록 합니다.
- 11) 1페이지의 다이어그램은 구성을 추상적으로 표현한 그림으로, 세부적인 구성은 아래의 요구사항을 만족시킬 수 있도록 합니다. (ex. 서브넷이 2개 이상 존재할 수 있습니다.)
- 12) 모든 리소스의 이름, 태그, 변수과 변수는 대소문자를 구분합니다.
- 13) 문제에서 주어지지 않는 값들은 AWS Well-Architected Framework 6 pillars를 기준으로 적절한 값을 설정해야 합니다.
- 14) 불필요한 리소스를 생성한 경우, 감점의 요인이 될 수 있습니다. (e.g. VPC 추가 생성)

### 3. Building an CI/CD Pipeline using ArgoCD and CodeSeries

Module: 12) CI/CD

#### 1. Bastion

- 채점을 위한 Bastion EC2를 생성하고, 정상적으로 EKS Cluster에 클러스터 관리자 권한으로 접근할 수 있어야 합니다.
- Bastion EC2는 채점을 위해 인스턴스 iam role에 AdministratorAccess 정책을 추가해야 합니다.
- Bastion EC2 설치 패키지: kubectl, argo rollouts CLI(kubectl plugin)

#### 2. 애플리케이션

- 애플리케이션은 8080 port에서 정상작동합니다.
- 애플리케이션은 EKS에 배포되어야 합니다.
- 애플리케이션은 **app** namespace에 배포되어야 합니다.

#### 3. CI/CD

- 주어진 애플리케이션 소스코드를 CodeCommit에 업로드해야 합니다.
- 소스코드가 업로드되는 CodeCommit repo의 이름은 **gwangju-application-repo**로 설정합니다.
- CodeCommit의 코드가 변경되면 3분 내로 새로운 애플리케이션이 배포되어야 합니다.
- 파이프라인은 컨테이너 이미지를 빌드해서 ECR에 업로드 하고, ArgoCD를 통해 애플리케이션이 업데이트 되어야 합니다.
- 애플리케이션은 Blue/Green 방식으로 업데이트 되어야 합니다.
- 업데이트로 인해 새로 생긴 애플리케이션이 준비가 완료되면 자동으로 기존 애플리케이션을 대체하도록 설정해야 합니다.
- 애플리케이션 업데이트가 완료되면 기존 애플리케이션 pod는 종료되어야 하며, 최대 2개의 이전 revision을 보관해야 합니다.
- Rollout 오브젝트의 이름은 **blue-green-app** 으로 설정합니다.

#### 4. ALB

- ALB를 통해 최신 애플리케이션에 접근할 수 있도록 구성해야 합니다.
  - ingress name : blue-green-ingress
  - ingress namespace : app
  - Application Load Balancer Name Tag : gwangju-blue-green-alb

## Application API Spec

Path	Method	Response body	Response Status Code
/	GET	"This is Blue Page"	200
/healthcheck	GET	"healthy"	200