

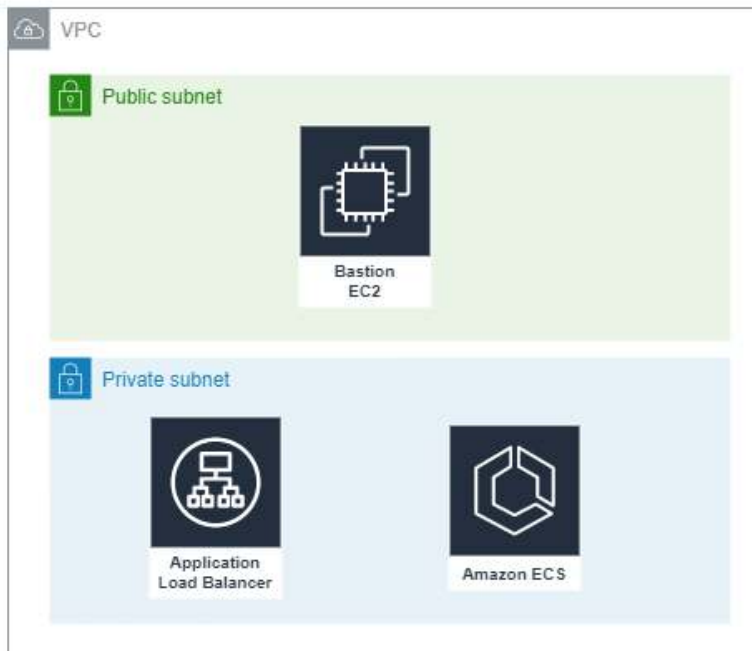
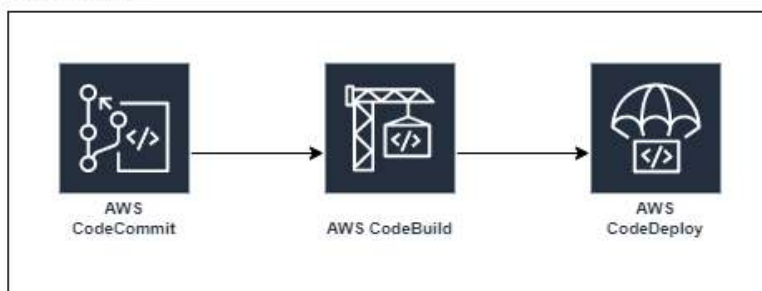
# 2022 경상남도 제57회 전국기능경기대회

직 종 명	클라우드컴퓨팅	과 제 명	Automation	과제번호	제2과제
경기시간	4시간	비 번 호		심사위원 확 인	(인)

## 1. 요구사항

어플리케이션의 안정적인 배포를 위한 CI/CD pipeline을 구성합니다. AWS Code series를 활용해 pipeline을 구성하며 컨테이너 형태로 ECS에 배포할 수 있어야 합니다.

Code Pipeline



### S/W stack

#### - AWS

- codecommit
- codebuild
- codedeploy
- codepipeline
- ecr
- ecs
- vpc
- route53
- ec2
- load balancer

#### - Application

- golang

## 2. 선수 유의사항

※ 다음 유의사항을 고려하여 요구사항을 완성하시오.

- 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 비용 제한이 존재합니다. 비용 제한 이상 사용 시 계정의 사용이 불가능할 수 있습니다.
- 6) 문제에 제시된 괄호는 변수를 뜻함으로 선수가 적절히 변경하여 사용하도록 합니다.
- 7) 문제의 효율을 위해 Security Group의 80/443 outbound는 anyopen하여 사용하도록 합니다.
- 8) 채점 시 Bastion EC2를 사용하오니 종료되어 불이익을 받지 않도록 합니다.
- 9) 과제 종료 시 진행 중인 테스트를 모두 종료하여 서버에 별도의 부하가 발생하지 않도록 합니다.
- 10) Region 지정이 필요한 서비스는 모두 ap-northeast-2 지역에 구성 하도록 합니다.
- 11) 고가용성 구성이 가능한 서비스는 생성한 AZ를 모두 활용하여 구성 하도록 합니다.
- 12) 채점은 Bastion 서버에 SSH 접근 후 시스템의 root 계정을 활용하여 채점 합니다. 접근 문제나 권한 문제가 있을시 채점에 불이익이 있을 수 있습니다.
- 13) ECS의 EC2 host에 대해 AMI type, Instance Type이 어떤값인지 채점 하진 않지만, 비용과 배포시간을 고려해 적절한 Instance Type을 찾아야 합니다. 즉 c5.large나 m5.xlarge 등 Type을 선택 하는것은 자유지만 배포 테스트 시 Instance Type 문제로 시간 초과할 경우 점수를 얻지 못합니다.
- 14) Code Deploy에서 배포 실패 테스트시 포트 변경이나 성공이 아닌 HTTP 응답코드, /health API 미존재 등으로 health check가 실패 하는지 확인 하여 채점을 진행 합니다. 채점지 기준으로 해당 정보가 맞지 않는 경우 Build 까지는 성공하고 Deploy 단계에서 배포가 되지 않는지 확인 합니다.
- 15) Code Deploy를 활용한 배포 채점 시 정상배포가 완료 되면 Deploy status가 Step1 ~ Step 4까지 모두 Succeeded, Traffic shifting progress가 Original 0%에 Replacement 가 100% 되어야 합니다.
- 16) 롤백 배포테스트 시 Deploy의 Install 단계에서 3분 이상 재시도 등으로 기다린 후 롤백이 진행 되어야 합니다. 즉 전체 배포가 7분안에 완료 되어야 함으로 Build 단계에서 4분 이상 소요되면 실패입니다.
- 17) 경기종료 전 ECS에 배포 되는 어플리케이션도 원본 코드로 배포 되어 있어야 합니다.

### 3. 네트워크 구성

VPC를 생성하여 클라우드 네트워킹을 구성 하세요. 10.10.0.0/16 네트워크를 사용합니다. vpc 이름은 skills-vpc입니다. HA를 고려하여 최소 3개의 az를 가지도록 VPC를 설계합니다. 아래 테이블의 Name은 서브넷의 Tag(Name)입니다. 테이블의 route table은 Route table의 Tag(Name)입니다. 테이블 정보를 활용해 Key를 Name으로, Value를 해당값으로 활용해 구성 합니다.

Name	Internet access	route table	subnet mask
skills-priv-a	NAT G/W	skills-priv-a-rt	10.10.0.0/24
skills-priv-b	NAT G/W	skills-priv-b-rt	10.10.1.0/24
skills-priv-c	NAT G/W	skills-priv-c-rt	10.10.2.0/24
skills-pub-a	Internet G/W	skills-pub-rt	10.10.10.0/24
skills-pub-b	Internet G/W	skills-pub-rt	10.10.11.0/24
skills-pub-c	Internet G/W	skills-pub-rt	10.10.12.0/24

### 4. APPLICATION

Golang으로 구현된 간단한 어플리케이션이 제공됩니다. /health API와 /v1/dummy API가 제공됩니다. 제공된 코드는 과제 진행 중 수정이 가능하나 채점을 위하여 경기 종료 전 제공된 원본을 업로드 해 두어야 합니다.

API	Response	기타
/health	code : 200, body : OK	
/v1/dummy	code : 200, body : BLUE	

### 5. ECS

EC2 base로 컨테이너가 동작 하도록 ECS를 구성합니다.

- cluster name : skills-cluster
- Task definition : skills-td
- Security Group : skills-ecs-sg
- Service : skills-svc

### 6. 로드밸런싱

ALB를 통해 로드밸런싱 합니다.

- ALB 이름 : skills-alb
- 타겟그룹 이름 : skills-tg1, skills-tg2
- ALB main listen port : 80

## 7. Code commit

제공된 코드를 코드커밋에 업로드 합니다. repository 이름은 skills-commit으로 생성하며 기본 브랜치 이름을 upstream으로 설정합니다.

## 8. Code Build

제공된 go lang app을 통해 컨테이너 이미지를 생성합니다. upstream branch에 코드가 커밋되면 자동으로 Docker image를 생성하여 skills-ecr 에 업로드 합니다. 업로드 시 이미지 태그 이름은 한국시간으로 현재 기준 날짜의 년-월-일.시.분.초 형식입니다. 예 (2022-06-26.09.01.59)

- Code build name : skills-build
- ECR name : skills-ecr(private)

## 9. Code Deploy

Build에서 생성된 이미지를 통해 새로운 버전의 어플리케이션을 배포할 수 있도록 합니다. Code Deploy의 Blue/Green 기능을 활용해 Blue/Green 배포가 가능해야 하며 health 상태가 되었을 때 한 번에 새로운 버전으로 전환 해야합니다. 만약 health check API 호출이 실패하면 새로운 버전이 배포되지 않도록 합니다. ECS의 컨테이너 헬스체크를 활용하여 실패된 버전이 배포되지 않도록 구성합니다. 문제가 있을 시 아래 그림처럼 Deployment의 Step1인 Install 단계에서 더 이상 진행이 되지 않아야 합니다.

Event	Duration	Status
BeforeInstall	less than one second	 Succeeded
Install	-	 In progress
AfterInstall	-	 Pending

- Application : skills-app
- Deployment Group : skills-dg
- Blue/Green에 사용하는 ALB target groups : skills-tg1, skills-tg2
- Cluster Name : skills-cluster
- Service : skills-svc

## 10. Code Pipeline

Code commit의 upstream branch 변경 시 자동으로 배포되도록 구성합니다. 위에서 생성한 Code commit, Code build, Code deploy를 각각 Source, Build, Deploy로 활용합니다. 채점 시 구성 전체에 별도의 승인 절차 없이 자동으로 배포가 완료 되거나 롤백 되는지 확인 합니다. 채점시 파이프라인 완료까지 7분이상 기다리지 않습니다.

- Pipeline name : skills-pipeline

## 11. DNS

Route53 private zone을 구성합니다. ALB의 값을 가리키는 레코드 하나도 생성하도록 합니다. Bastion 서버에서 해당 DNS를 통하여 ALB에 호출합니다.

- DNS Name : skills.local (Private zone)
- Record Name : app.skills.local (ALB address)

## 12. Bastion

서버 접근 및 채점을 위해 Bastion 서버를 구성합니다. 해당 서버에 접근 불가할 시 채점이 불가함으로 반드시 SSH를 통한 접속과 권한 문제가 없도록 합니다. Bastion은 awscli 입력 시 Admin Policy에 상응하는 권한을 가지고 있어야 합니다. Bastion 서버에서 문제에서 구성한 ALB에 접근 가능해야 합니다. 재시작 되어도 IP가 변경 되지 않아야 합니다.

- EC2 type : t3.small
- 이미지 : Amazon Linux2
- 설치 패키지 : awscli, jq, curl
- Tag : Name=skills-bastion2