

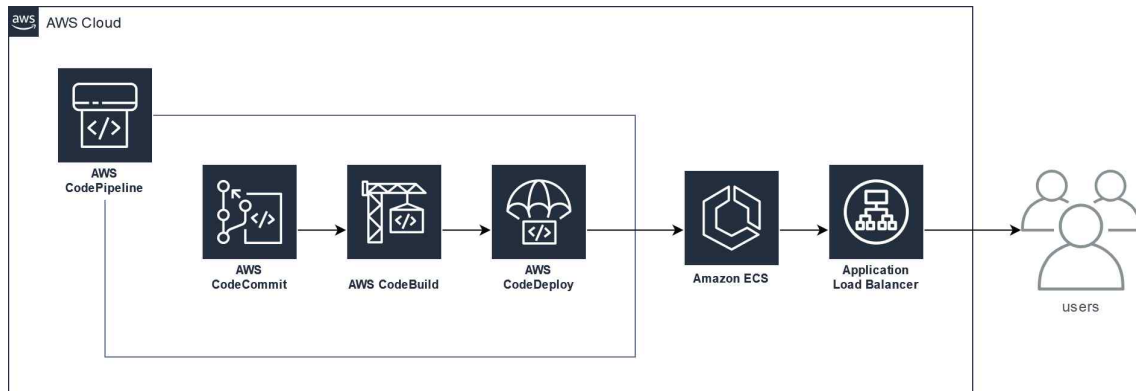
2024년도 전국기능경기대회 과제

직 종 명	클라우드컴퓨팅	과제명	Automation	과제번호	제2과제
경기시간	4시간	비번호		심사위원 확 인	(인)

1. 요구사항

당신은 주어진 요구사항과 클라우드의 설계원칙인 고가용성, 확장성, 비용, 보안 등을 고려하여 인프라를 구축합니다.

다이어그램



Software Stack

AWS	개발언어/프레임워크
<ul style="list-style-type: none"> - VPC - ECS - ELB - EC2 - CodeCommit - CodeBuild - CodeDeploy - CodePipeline 	<ul style="list-style-type: none"> - HTML - Docker

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 비용제한이 존재하며, 이보다 더 높게 과금될 시 계정 사용이 불가능할 수 있습니다.
- 6) 문제에 제시된 괄호박스는 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 발생 하지 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, 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. 네트워크 구성

클라우드 인프라 구성을 위해 아래 요구사항을 참고하여 VPC를 구성합니다. 서브넷 이름 뒤의 알파벳은 Availability Zone을 의미합니다.

VPC

- Name : wsi-vpc
- CIDR : 10.150.0.0/16

App Subnet A

- Name : wsi-app-a
- CIDR : 10.150.0.0/24
- Gateway : wsi-natgw-a
- Route Table : wsi-app-a-rt

App Subnet B

- Name : wsi-app-b
- CIDR : 10.150.1.0/24
- Gateway : wsi-natgw-b
- Route Table : wsi-app-b-rt

Public Subnet A

- Name : wsi-public-a
- CIDR : 10.150.10.0/24
- Gateway : wsi-igw
- Route Table : wsi-public-rt

Public Subnet B

- Name : wsi-public-b
- CIDR : 10.150.11.0/24
- Gateway : wsi-igw
- Route Table : wsi-public-rt

4. Bastion 서버

EC2를 활용해 Bastion 서버를 구성합니다. bastion 서버의 접근을 위해서 SSH 프로토콜을 사용합니다. 해당 인스턴스에서 AWSCLI를 사용하면 AWS 내 모든 리소스에 대해 접근할 수 있어야 하며, 채점을 위해 사용되는 패키지들은 권한 등의 문제가 없어야 합니다.

Bastion 서버는 채점을 위해서 사용됩니다. 잘 못 구성하였을 경우 모든 채점 항목에서 불이익을 받을 수 있으니 주의합니다.

- Instance type : t3.small
- 설치 패키지 : awscli, curl, jq
- Tag : Name=wsi-bastion

5. 웹 어플리케이션

제공되는 어플리케이션은 1개입니다. HTML로 개발된 단일 페이지 어플리케이션입니다. NGINX를 통해 HTML 파일을 배포해야 합니다.

6. 컨테이너 오케스트레이션

어플리케이션은 ECS 기반의 EC2 인스턴스에서 운용합니다. EC2 인스턴스는 App Subnet에서 운용해야 합니다.

- ECS Cluster Name : wsi-ecs
- ECS EC2 Instance Type : t3.medium
- ECS Service Name : wsi-ecs-s

7. 로드 밸런싱

ECS에서 운영 중인 어플리케이션을 외부에서 접근할 수 있도록 ALB를 구성합니다.

- Load Balancer Type : Application Load Balancer
- Load Balancer name : wsi-alb
- Load Balancer Scheme : internet-facing
- Load Balancer Listener Port : 80

8. Version Control

CodeCommit을 버전 관리와 CI/CD에 사용합니다. 제공 자료를 압축 해제한 후, Repository의 최상단(/)에 업로드합니다. Repository 내부에는 제공 자료 이외의 파일이 있어도 무관하나, 채점 전 반드시 제공된 자료의 내용이 원본 상태 그대로 존재해야 합니다.

- Repository Name : wsi-commit
- Default Branch : main

9. Continuous Integration

CodeBuild를 CI에 사용합니다. NGINX 기반의 HTML 어플리케이션을 빌드합니다.

- Project Name : wsi-build

10. CI/CD Pipeline

CodeCommit의 main branch에 변경 사항이 발생할 시 자동으로 CodePipeline이 동작하도록 구성해야 합니다. Source, Build의 총 2개 Stage는 필수적으로 구성되어야 하며, CodeCommit에 변경 사항이 발생한 이후 15분 내로 어플리케이션 배포가 완료되어야 합니다.

- Pipeline Name : wsi-pipeline
- Source Stage : CodeCommit wsi-commit main branch
- Build Stage : CodeBuild wsi-build