

ExamForge (2025.10.27 - 2025.11.02)



프로젝트 개요

AWS 자격증 시험 대비 실전 모의고사 플랫폼입니다. AWS MediaLive로 RTMP 입력을 HLS로 변환하고, CloudFront CDN을 통해 전세계 200+ 엣지 로케이션으로 HTTPS 배포합니다. Terraform IaC로 전체 인프라를 자동화하여 빠른 배포 가능 간편한 유지보수.

핵심 기능

- **ALB** - 트래픽 분산 및 고가용성, HTTPS 인증서 통합 관리
- **ECS** - 컨테이너 오케스트레이션 배포 자동화 및 관리 용이
- **Fargate** - ECS에 서비스 Fargate 사용 오버헤드 감소
- **GitHub** - 버전 관리를 통해 장애시 빠른 복구
- **Terraform** - 인프라 구축 자동화 및 유지보수

기술 스택

FrontEnd

BackEnd

CI/CD

Terraform (IaC)

AWS ECS

AWS Fargate

AWS DynamoDB

AWS CloudWatch

프로젝트 스크린샷

팀원 역할 및 핵심 책임
Key Roles & Responsibilities

박대복
정적 컨텐츠

담당 업무:

- 웹페이지 및 컨텐츠 제작
(Front-End Development)
- GitHub Repository 통합 및 관리
- CI 파이프라인 구축 및 테스트 주도
(Continuous Integration)

장경진
인프라

담당 업무:

- AWS 클라우드 인프라 구축 및 운영
- Github Actions 기반 CI 파이프라인
구축 및 자동 배포 관리
- 배포 전환 수립 및 실행
- 프로젝트 성과 발표 자료 제작

개발 트랙

인프라 트랙

STEP 1 웹 컨텐츠 제작

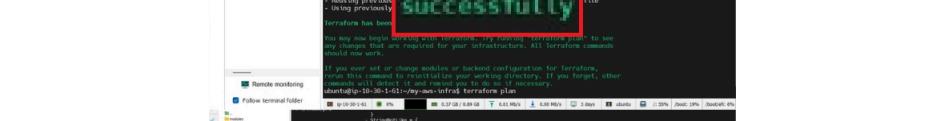
STEP 1 테라폼 인프라 구성

STEP 2 CI 파이프라인 구축

STEP 2 CD 파이프라인 구축

STEP 3 통합 테스트 및 배포

A screenshot of a mobile application's login screen. The background is dark grey. In the center is a white rectangular box containing the text "로그인 성공" (Login Success) in bold black font. To the right of this text is a blue rounded rectangle button with the word "확인" (Confirm) in white. Below this, the word "로그인" (Login) is centered in a smaller white box. Underneath the "로그인" box are two empty input fields, each with a placeholder icon on the left. Below these fields is a small ellipsis (...). At the bottom of the white box is a large, dark grey rectangular button with the word "로그인" (Login) in white. At the very bottom of the screen, there is a small, light blue footer bar with the text "계정이 없으신가요? 회원가입" (Don't have an account? Sign up) in white.



The terminal window shows the output of a Terraform plan command:

```
aws_lambda_function "my-new-lambda" {
  function_name = "my-new-lambda"
  role          = "arn:aws:iam::123456789012:role/service-role/lambdaBasicExecutionRole"
  handler       = "app.handler"
  runtime       = "python3.7"
  code          = "file:///tmp/app.zip"
}

# module "lambda" is already up-to-date.

# Changes proposed by this plan:
#   to add: 0
#   to change: 0
#   to destroy: 0

Plan: 0 to add, 0 to change, 0 to destroy
```

The status bar at the bottom indicates "Terraform successfully applied".

The screenshot shows a CI/CD pipeline interface with a log titled 'deploy'. The log details the execution of various steps: Set up job, Set Dynamic Variables, Checkout repository, Configure AWS credentials, Login to Amazon ECR, Build, Tag, Push Docker Image, Generate Fargate Task Definition with ALB Health Check, Deploy to ECS Fargate, Login to Amazon ECR, Configure AWS credentials, Checkout repository, and Complete job. All steps are marked as completed successfully.

The screenshot shows a database table titled '테이블: User-Data-Dev - 반환된 항목 (1)'. It displays one row with columns: UserID (문자열), createdAt, email, name, and password. The first column has a green checkmark icon. The table shows a single record with values: 785bec5d-442d-4cf... (UserID), 2025-11-0... (createdAt), aaa@aaa (email), aaa (name), and \$2b\$10\$foUexLJ8ZaE4a5AnsJ6... (password).

성과

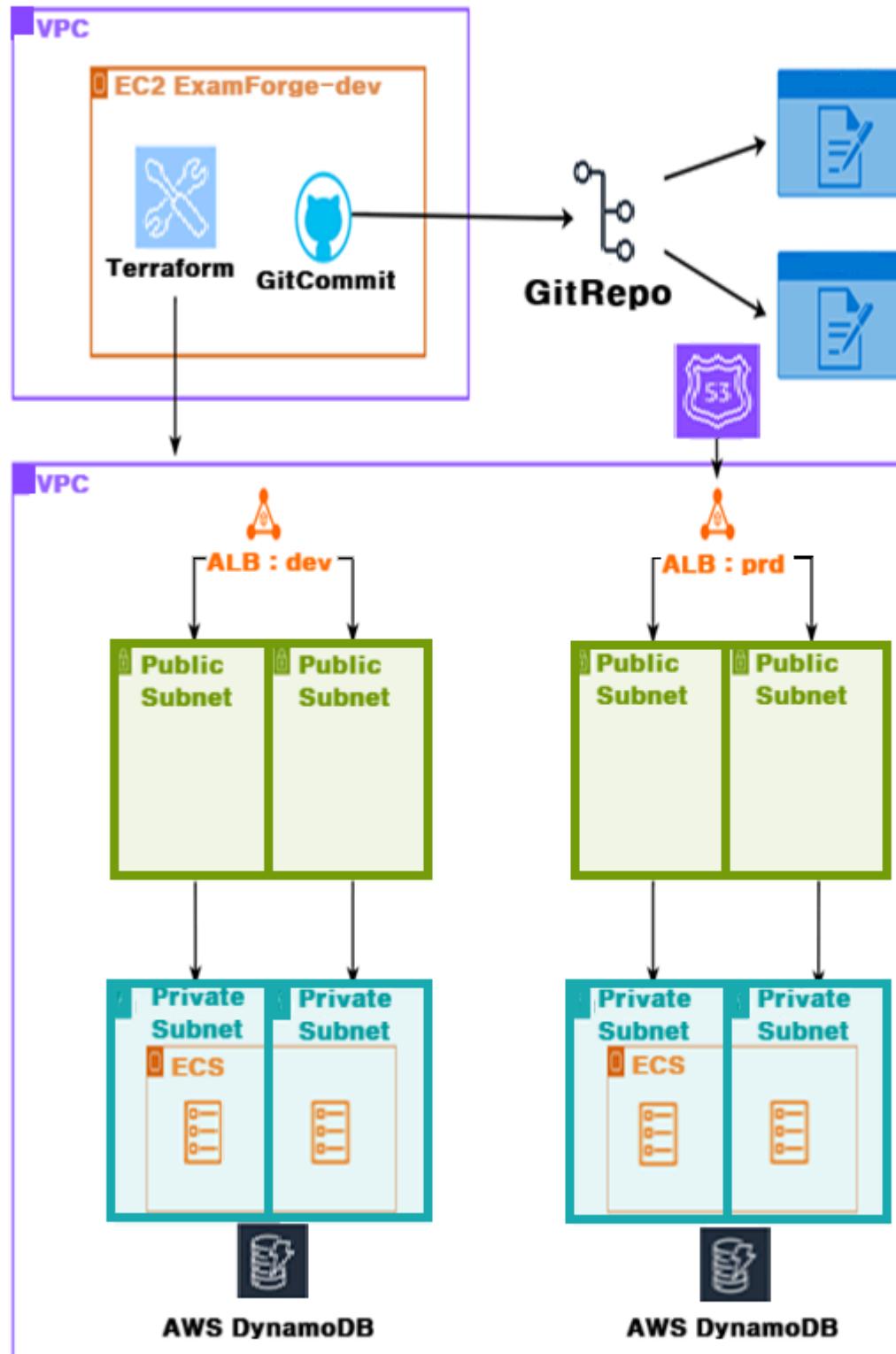
- CI/CD 파이프라인
- 인프라 자동화
- 팀원과의 업무 분배 경험
- GitHub 버전 관리에 대한 이해

1 / 2

ExamForge - 인프라 설계

시스템 아키텍처

ap-northeast-2



테라폼 트리구조

```
..terraform.lock.hcl
├── dynamodb.tf                                # DynamoDB
├── iam_ecs.tf
├── iam_oidc.tf                                 # GitHub Action
├── main.tf                                       # 메인 설정 파일
├── outputs.tf
├── providers.tf
├── route53.tf                                   # DNS 설정
├── terraform.tfvars
└── variables.tf

└── modules
    ├── ecs-cluster                             # ECS 클러스터
    │   ├── main.tf
    │   ├── outputs.tf
    │   └── variables.tf

    └── vpc                                     # VPC 네트워크
        ├── .terraform.lock.hcl
        ├── main.tf
        ├── outputs.tf
        └── variables.tf
```

상세 구현 내용

FrontEnd

정적 컨텐츠 제작 HTML, CSS, JavaScript (박대복)

BackEnd

Node.js를 이용한 Docker 이미지 제작을 통한 서버 구현(박대복)

Terraform (IaC)

코드를 통한 인프라 아키텍처 제작 (장경진)

CI/CD 파이프라인

GitHub repo에 파일 업로드 및 수정 시 자동 배포가 가능하도록 파이프라인 설계 (장경진)

PPT 제작 및 프로젝트 서류화

발표자료(PPT) 제작 및 프로젝트 문서화 (장경진)

비용 분석

테스트 비용

일일: ~\$2.68 / 중지 상태: ~ \$0.5/일일

월 비용 (일 24시간 기준)

\$89/월