**SK네트웍스 Family AI과정 3기  
 모델배포 시스템 구성도**

**□ 개요**

* 산출물 단계 : 모델배포
* 평가 산출물 : 시스템 구성도
* 제출 일자 : 2024-12-25
* 깃허브 경로 : <https://github.com/SKNETWORKS-FAMILY-AICAMP/SKN03-FINAL-6Team>
* 작성 팀원 : 최연규

|  |  |  |
| --- | --- | --- |
| **개요** | * 모델 배포 시스템은 AWS 클라우드 환경을 기반으로 구축되었습니다. * CI/CD 파이프라인을 통해 지속적인 배포와 자동화된 운영을 지원합니다. * FastAPI 기반 모델 API와 Spring Boot 백엔드 서비스를 ECS 위에 배포하여 안정성과 확장성을 확보하였습니다. * 데이터 저장소로는 RDS(MySQL)와 Milvus를 사용합니다. * 프론트엔드는 React를 활용하여 CloudFront와 S3에 배포되었습니다. | |
| **구성 요소** | CodePipeline | 지속적인 통합 및 자동 배포 도구로, 소스 코드 변경 시 빌드 및 배포 자동화. |
| ECR | 도커 이미지 저장소로, FastAPI, Spring Boot, React 애플리케이션 이미지를 저장. |
| ECS | 컨테이너화된 애플리케이션을 실행하며, Fargate로 오토스케일링 지원. |
| ALB (로드 밸런서) | HTTP 요청을 각 컨테이너 서비스로 분산 처리. |
| RDS (MySQL) | 관계형 데이터베이스 서비스로 사용자 및 모델 관련 데이터를 관리. |
| Milvus | 벡터 데이터 저장 및 검색을 위한 벡터 데이터베이스. |
| S3 | 정적 파일(React 프론트엔드)을 저장 및 제공. |
| CloudFront | S3와 연계하여 프론트엔드 콘텐츠를 사용자에게 빠르게 배포. |
| Route 53 | 도메인 이름 관리 및 트래픽 라우팅. |
| Certificate Manager | HTTPS 연결을 위한 SSL 인증서 관리. |
| Parameter Store | 중요한 애플리케이션 설정 (예: 비밀번호, API 키) 안전하게 저장. |
| **데이터 흐름** | CI/CD 파이프라인 | 개발자가 소스 코드를 수정하여 GitHub에 푸시하면, CodePipeline이 이를 감지하고 빌드 및 배포 과정을 자동으로 실행. 빌드된 컨테이너 이미지는 ECR에 저장. |
| 모델 및 서비스 배포 | FastAPI 기반의 모델 API와 Spring Boot 서비스는 ECS Fargate를 통해 컨테이너로 실행. 요청은 ALB를 통해 적절한 서비스로 라우팅. |
| 데이터 관리 | 모델 및 사용자 데이터는 RDS(MySQL)에 저장되며, 벡터 데이터는 Milvus에서 관리. |
| 프론트엔드 제공 | React로 구성된 프론트엔드는 S3에 업로드되고, CloudFront를 통해 사용자에게 빠르게 배포. |
| HTTPS 및 도메인 관리 | Certificate Manager를 통해 HTTPS 연결을 제공하며, Route 53을 사용해 도메인 이름을 관리. |

