

멋쟁이사자처럼 DevOps 7기 · 3차 프로젝트 오리엔테이션

하이브리드 클라우드 프로젝트 진행 가이드

2026.10.05(월) ~ 11.13(금) · 6주 스프린트 · 4개 팀



면접 질문 다섯 개에 직접 한 일로 답하기

기능을 많이 만드는 프로젝트가 아닙니다. 하나의 앱을 온프레미스에서 클라우드까지 옮기며 배포를 자동화합니다.



서버 구성

리눅스와 쿠버네티스를 직접 구성해 봤나요?



배포 자동화

CI/CD로 배포를 자동화해 봤나요?



IaC

인프라를 Terraform 코드로 관리해 봤나요?



장애 복구

롤백과 백업·복구를 실제로 해 봤나요?

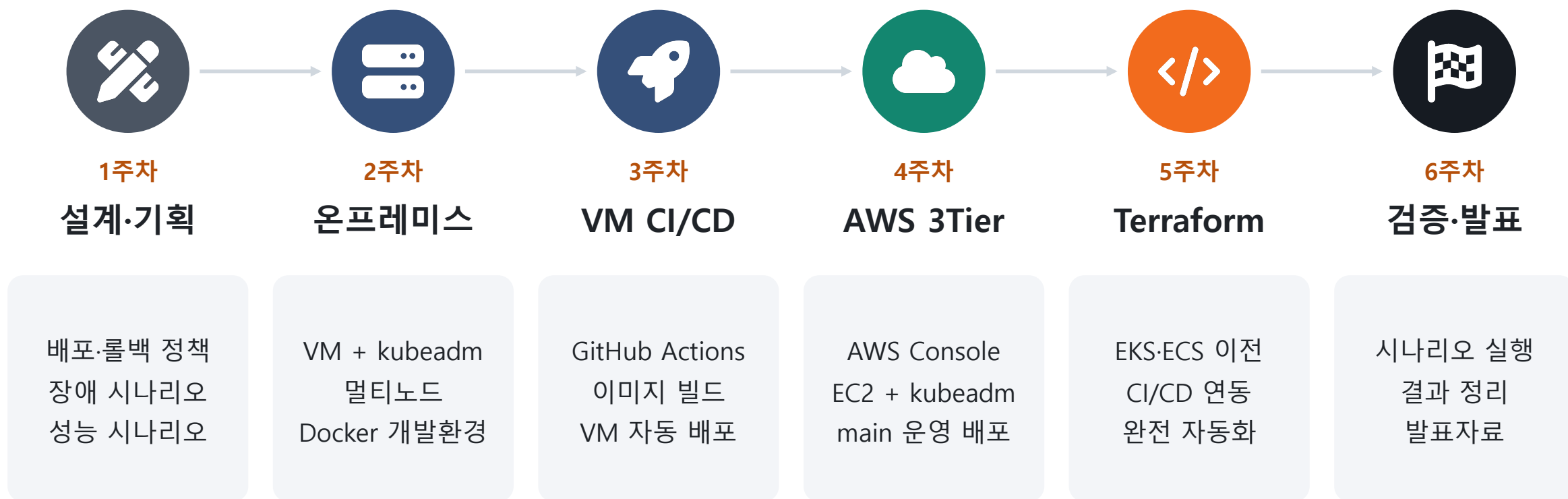


성능 개선

부하 테스트로 성능을 숫자로 개선해 봤나요?

6주 뒤, 이 질문에 이슈·PR·위키·그래프를 근거로 답할 수 있어야 합니다.

같은 앱을 환경만 바꿔 세 번 배포합니다



매 단계마다 "이전에 손으로 하던 일"을 자동화로 바꾸는 것이 핵심입니다.

6주 마일스톤 — 월요일 시작, 금요일 마감



1주차는 팀 활동으로 진행하는 기획·설계 스프린트 **금요일 16:00** 팀장 마감 미팅에서 결과 보고

역할별 마일스톤 — 4역할 × 6주

팀당 4명 · 역할마다 6개 · 화·목 배포 · 금 마감

	온프레미스 (ON)	클라우드 (CL)	백엔드 (BE)	프론트엔드 (FE)
1주차 10/05–10/09	ON1 온프레미스 설계 10/07 VM·네트워크 설계	CL1 AWS 3Tier 설계 10/08 구성도·비용표	BE1 API·데이터 설계 10/07 기준 앱 확정	FE1 화면·흐름 설계 10/08 정적 분리 결정
2주차 10/12–10/16	ON2 VM·kubeadm 구축 10/15 K8s 클러스터 배포	CL2 계정·네트워크 기반 10/13 콘솔로 VPC 구성	BE2 컨테이너 대응 10/14 헬스체크 추가	FE2 정적 프론트 분리 10/15 Nginx로 배포
3주차 10/19–10/23	ON3 온프레미스 CI/CD 10/20 develop 자동 배포	CL3 데이터 계층·CI 권한 10/20 RDS·Redis 접속	BE3 테스트·DB 마이그레이션 10/21 Flyway V1	FE3 프론트 CI/CD 10/22 파이프라인 합류
4주차 10/26–10/30	ON4 개발 환경 운영 10/27 모니터링 구축	CL4 AWS 3Tier 운영 배포 10/27 main → EC2 배포	BE4 운영 환경 대응 10/29 메트릭 노출	FE4 클라우드 웹 계층 10/27 CloudFront·HTTPS
5주차 11/02–11/06	ON5 하이브리드 통합 11/03 한 차트로 두 환경	CL5 Terraform EKS/ECS 11/05 무인 배포	BE5 무중단·확장 대응 11/05 요청 실패 0 측정	FE5 배포 자동화·성능 11/03 캐시 무효화 자동
6주차 11/09–11/13	ON6 장애·복구 검증 11/10 etcd 복구 리허설	CL6 장애·확장 검증 11/09 HPA 확장 관찰	BE6 성능·장애 검증 11/09 k6 측정	FE6 사용자 관점 검증 11/12 데모 리허설

※ 1주차: 10/05(월) 대체공휴일(개천절) · 10/07(수) 공식 시작 · 10/09(금) 한글날 · 11/13(금) 최종 발표

온프레미스 마일스톤 ON1~ON6

개발·검증 환경 (develop 배포)

ON1 1주차 · 10/05~10/09 온프레미스 설계

- 수 VM 구성·네트워크 설계
- 목 런타임·CNI 선택, 백업 대상
- 목 온프레미스 배포·롤백 정책

완료 위키에 온프레미스 설계 등록

ON2 2주차 · 10/12~10/16 VM·kubeadm 구축

- 월 VM 3대 설치·공통 설정
- 화 compose로 앱 첫 배포
- 수 kubeadm·CNI·metrics-server
- 목 매니페스트·Ingress 배포

완료 노드 3대 Ready, Pod 자동 복구

ON3 3주차 · 10/19~10/23 온프레미스 CI/CD

- 월 self-hosted runner 등록
- 화 GHCR 이미지 → 자동 배포
- 수 롤백 리허설 (BE·FE와)
- 목 Secret 정리·실패 알림

완료 develop 자동 배포, 롤백 기록

ON4 4주차 · 10/26~10/30 개발 환경 운영

- 월 develop→온프렘, main→클라우드
- 화 Prometheus·Grafana
- 수 etcd·MySQL 정기 백업
- 목 대시보드·알림 점검

완료 지표 대시보드, 백업 파일 확인

ON5 5주차 · 11/02~11/06 하이브리드 통합

- 월 Helm 차트 통일 (온프렘 values)
- 화 한 차트로 두 환경 배포
- 수 온프레미스 백업 → S3
- 목 노드 점검 절차(drain)

완료 한 차트로 배포, S3 백업 도착

ON6 6주차 · 11/09~11/13 장애·복구 검증

- 월 워커 노드 다운 시험
- 화 etcd 복구 리허설
- 수 MySQL 백업 복원
- 목 온프렘 vs 클라우드 비교

완료 시나리오별 복구 시간 등록

클라우드 마일스톤 CL1~CL6

운영 환경 (main 배포)

CL1

1주차 · 10/05-10/09

AWS 3Tier 설계

- 수 VPC·서브넷 설계 (AZ 2개)
- 목 3Tier 구성도
- 목 비용 추정·IAM 최소 권한

완료 위키에 구성도·비용표 등록

CL2

2주차 · 10/12-10/16

계정·네트워크 기반

- 월 MFA·IAM·예산 알림
- 화 콘솔로 VPC·서브넷
- 수 라우팅·IGW·NAT
- 목 보안그룹 계층 설계

완료 VPC 기본망, 예산 알림 수신

CL3

3주차 · 10/19-10/23

데이터 계층·CI 권한

- 월 RDS·ElastiCache 생성
- 화 VPC 내부 접속 확인
- 수 Actions OIDC → AWS 역할
- 목 CI 재사용 구조 합의

완료 DB 접속, Actions AWS 인증

CL4

4주차 · 10/26-10/30

AWS 3Tier 운영 배포

- 월 EC2에 kubeadm 클러스터
- 화 main 머지 → EC2 배포
- 수 ALB·HTTPS·보안그룹
- 목 RDS 스냅샷·로그 수집

완료 운영 주소에서 한 바퀴 동작

CL5

5주차 · 11/02-11/06

Terraform EKS/ECS

- 월 state 백엔드·VPC·DB 코드화
- 화 EKS(ECS)·ECR·Secrets
- 수 HPA·노드 오토스케일
- 목 PR→plan/apply→배포

완료 apply로 재생성, 수동 작업 0

CL6

6주차 · 11/09-11/13

장애·확장 검증

- 월 k6 부하 중 HPA 관찰
- 화 RDS 스냅샷 복원
- 수 노드 장애 가정 시험
- 목 비용 정리·destroy 리허설

완료 확장·복원·비용 위키 등록

백엔드 마일스톤 BE1~BE6

게시판을 운영 가능한 API로

BE1

1주차 · 10/05-10/09

API·데이터 설계

- 수 기준 앱 확정·테스트 초록
- 목 API 명세·ERD 위키 등록
- 목 환경 변수·헬스체크 합의

완료 위키에 API·ERD 등록

BE2

2주차 · 10/12-10/16

컨테이너 대응

- 월 로그 stdout·설정 외부화
- 화 compose 배포 지원
- 수 Actuator probe·정상 종료
- 목 CORS·쿠키(SameSite)

완료 probe 통과, 분리 화면 로그인

BE3

3주차 · 10/19-10/23

테스트·DB 마이그레이션

- 월 통합 테스트 (MySQL·Redis)
- 화 CI 테스트 = 머지 조건
- 수 schema.sql → Flyway V1
- 목 /actuator/info 커밋 SHA

완료 CI 게이트, 마이그레이션 자동

BE4

4주차 · 10/26-10/30

운영 환경 대응

- 월 RDS·ElastiCache 연결 검증
- 화 운영 배포 지원
- 수 풀 크기·스케줄러 락·캐시
- 목 Prometheus 메트릭

완료 레플리카 2개 이상 정상 동작

BE5

5주차 · 11/02-11/06

무중단·확장 대응

- 월 무상태 점검 (HPA 대비)
- 화 AI·외부 호출 타임아웃
- 수 하위 호환 스키마 변경
- 목 롤링 배포 중 실패율 측정

완료 롤링 배포 중 요청 실패 0

BE6

6주차 · 11/09-11/13

성능·장애 검증

- 월 k6 시나리오 4종 측정
- 화 Redis·AI 중단 시 동작
- 수 병목 1개 개선
- 목 재측정·발표 자료

완료 개선 전→후 수치 등록

프론트엔드 마일스톤 FE1~FE6

화면 분리와 웹 계층

FE1

1주차 · 10/05-10/09

화면·흐름 설계

- 수 화면 목록·사용자 흐름
- 목 화면↔API 매핑표
- 목 정적 프론트 분리 결정

완료 위키에 화면·흐름 등록

FE2

2주차 · 10/12-10/16

정적 프론트 분리

- 월 frontend/ 폴더 분리
- 화 api.js 환경별 주소
- 수 Nginx 컨테이너·/api 프록시
- 목 VM 클러스터에 배포

완료 분리 화면에서 로그인·글쓰기

FE3

3주차 · 10/19-10/23

프론트 CI/CD

- 월 린트·포맷 검사
- 화 Nginx 이미지 빌드 (SHA)
- 수 캐시 무력화 (파일 버전)
- 목 배포 파이프라인 합류

완료 develop push 시 자동 배포

FE4

4주차 · 10/26-10/30

클라우드 웹 계층

- 월 S3에 정적 파일 배포
- 화 CloudFront·HTTPS·도메인
- 수 쿠키·CORS 백엔드 합의
- 목 4상태 화면 운영 점검

완료 운영 도메인에서 전 화면 동작

FE5

5주차 · 11/02-11/06

배포 자동화·성능

- 월 S3 업로드 자동화
- 화 CloudFront 무효화 자동
- 수 Terraform에 S3·CloudFront
- 목 Lighthouse 측정·개선

완료 push→무효화까지 무인 실행

FE6

6주차 · 11/09-11/13

사용자 관점 검증

- 월 API 장애 시 화면 동작
- 화 AI 실패해도 글 표시
- 수 Lighthouse 전→후 비교
- 목 데모 시나리오 리허설

완료 데모 리허설 1회 완주

하루 일정



09:10 – 09:40

오전 스크럼 · 30분

- 어제 한 일
- 오늘 할 일
- 막힌 것

보드를 보며 한 사람씩 짧게



10:00 – 11:50

자율 시간 · 110분

- 강의
- 모의면접
- 개인·팀 학습

각자 계획에 따라 활용



17:00 – 18:00

프로젝트 회고 · 60분

- 오늘 결과 공유
- 막힌 문제 정리
- 내일 계획

결정 사항은 회의록 이슈
[MEETING]으로 기록

WEEKLY

한 주 시간표

	월	화	수	목	금
09:10	오전 스크럼	오전 스크럼	오전 스크럼	오전 스크럼	오전 스크럼
10:00	자율 시간	자율 시간	자율 시간	자율 시간	자율 시간
14:00	team01 미팅	team02 미팅	team03 미팅	team04 미팅	
16:00	팀장 주간 미팅				팀장 마감 미팅
17:00	회고	회고	회고	회고	회고
배포		배포일		배포일	

※ 1주차 공휴일: 10/05(월) 대체공휴일(개천절) · 10/09(금) 한글날 — 1주차 팀장 주간·마감 미팅은 별도 공지

배포는 화·목에만 합니다

화 · 목

develop → main PR 머지
→ 배포 환경 반영

다른 날은 develop에서
개발하고 검증만

주차별 배포 환경

1주차는 기획·설계로 배포 없음



2~3주차

온프레미스 VM 클러스터



4주차

AWS EC2 클러스터



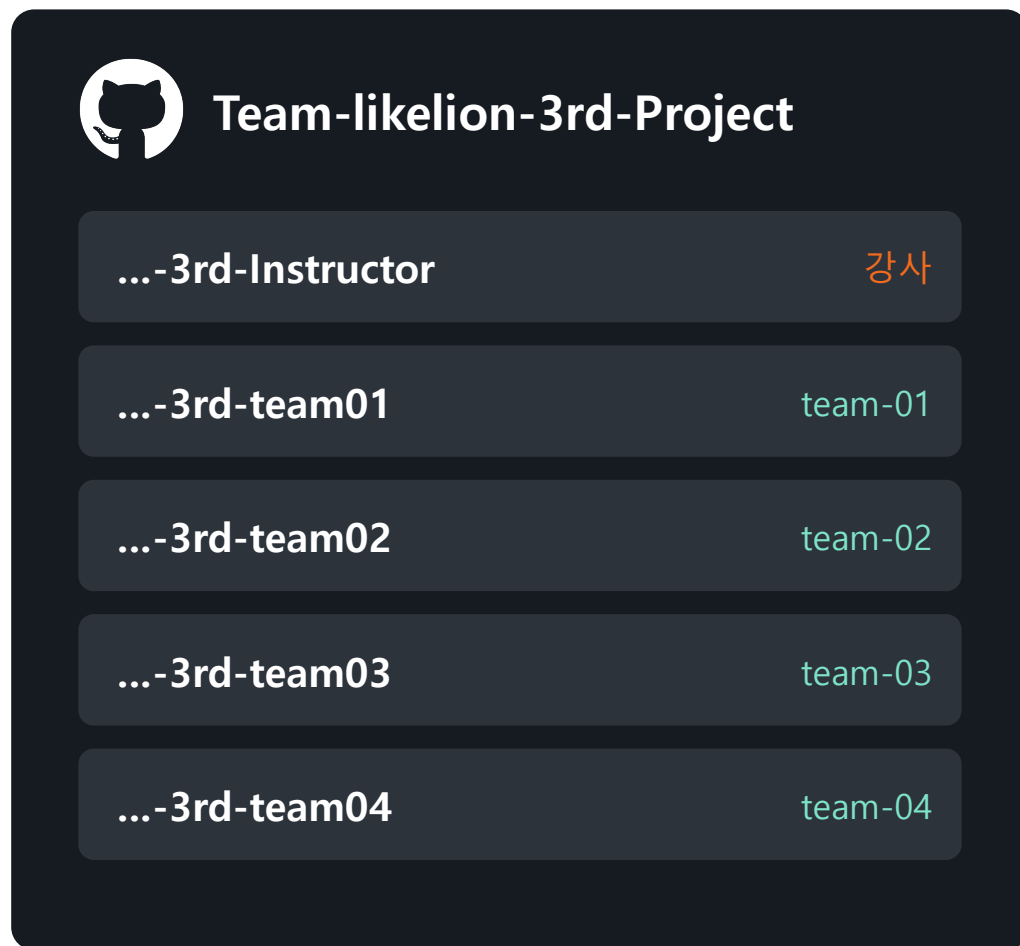
5~6주차

Terraform으로 만든 EKS·ECS

배포 규칙

- main PR은 승인 2명 + 코드 오너 승인 — 리뷰는 배포일 전에 미리
- 배포 결과는 그날 17:00 회고에서 공유
- 실패하면 롤백 정책대로 되돌리고 [TS] 이슈로 기록

GitHub 조직 하나, 팀마다 레포 하나



이슈 템플릿 5종

FEAT · BUG · TS · DOCS · MEETING



PR 템플릿

이슈 연결 · 체크리스트 · 리뷰 포인트



라벨 26개

유형 · 영역 · 우선순위 · 상태 · 역할(role/*)



마일스톤 M1~M6

6주 로드맵 그대로



진행보드

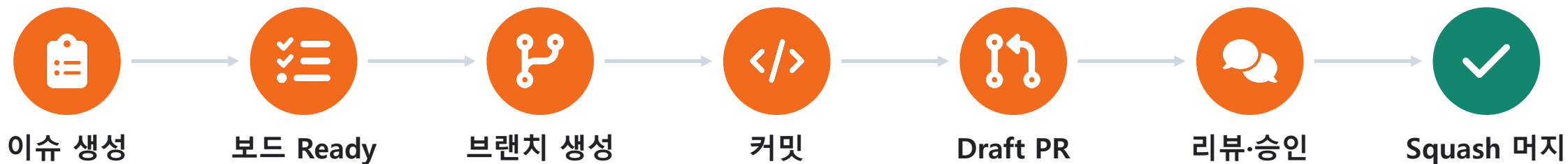
칸 5개 · 역할·스프린트 필드 · 템플릿 일감 100장



위키 + 가이드

설계·운영 문서 뼈대, CONTRIBUTING, GUIDE

작업 한 사이클



이름 규칙

브랜치	<code>feature/#12-login-api</code> <code>fix/#15-null-check</code>
커밋	feat: 로그인 API 구현 (#12)
PR 본문	Closes #12 → 머지 시 이슈 자동 종료

브랜치 보호

브랜치	직접 push	머지 조건
develop	불가	승인 1명
main	불가	승인 2명 + 코드 오너 승인

- 공통: 리뷰 코멘트를 모두 Resolve해야 머지
- Squash 머지만 허용 · 머지된 브랜치 자동 삭제

기록이 곧 포트폴리오입니다

면접에서 "그걸 정말 직접 했나요?"에 답할 수 있는 건 기록뿐입니다.



회의록

매일 회고 · 주간 미팅

회의록 이슈
[MEETING]



트러블슈팅

막힐 때마다

이슈 [TS] →
해결 후 위키



설계 근거

1주차 · 구조 변경 시

위키 아키텍처
배포 가이드



검증 결과

6주차

위키 장애 시나리오
성능 테스트



내 기여

매일

내 이름의
이슈 · PR · 리뷰

- 결정되지 않은 것은 "보류"로 — 추측해서 적지 않기
- 에러는 로그 원문·캡처로 — "안 됐다"는 기록이 아님
- PR 리뷰 코멘트도 기여 — 근거 있는 코멘트 남기기

절대 하면 안 되는 것



비밀값 커밋

AWS 키·비밀번호·env·*.pem은 절대 올리지 않습니다. 레포는 공개입니다. 올렸다면 즉시 키를 폐기하고 재발급하세요.



AWS 리소스 방치

실습 후 EC2·NAT Gateway·EKS 등 과금 리소스를 정리합니다. Terraform으로 만든 것은 terraform destroy.



develop·main 직접 push

막혀 있지만 우회하지 마세요. 모든 변경은 브랜치 → PR → 리뷰 → 머지.



혼자 오래 막혀 있기

30분 넘게 진전이 없으면 help-wanted 이슈를 만들고 스크럼·회고에서 공유합니다.

첫날 체크리스트



개인

- ☐ 조직 Team-likelion-3rd-Project 초대 수락
- ☐ 팀 레포 clone, git user.name·email 설정
- ☐ CONTRIBUTING.md, docs/GUIDE.md 읽기

TIP git 이메일은 GitHub 계정 이메일과 같게 설정해야 잔디(기여 기록)가 남습니다.



팀 (팀장 주도)

- ☐ 역할 4개 담당자 정하기 (ON·CL·BE·FE)
- ☐ CODEOWNERS에 팀원 GitHub ID 채우기
- ☐ README.md의 {...} 채우기
- ☐ 진행보드 Auto-add to project 켜기
- ☐ M1 이슈 등록 — 담당자·role 라벨
- ☐ 첫 회의록 이슈 작성 (그라운드 룰)

기억할 세 가지

이슈로 시작하고,
화·목에 배포하고,
모든 것을 기록합니다.

질문은 오전 스크럼 · 팀 진행 미팅 · help-wanted 이슈로