DevOps를 더 잘하기 위한 Datadog 플랫폼 활용법 💡

정영석

기술팀 리더

이누리

Sales Engineer

DATADOG

오늘 할 이야기는..

우리는 **DevOps**를 잘 하고 있을까?

DevOps 성숙도를 확인하기 위한 **DORA**에 대해 이야기 해보겠습니다

Agenda

DevOps 성숙도를 높이기 위한 노하우를 살펴봅니다 💡

1. DORA는 무엇일까요?
2. 배포 빈도와 배포 리드 타임
3. 변경 실패율
4. 평균 복구 시간

DATADOG

많은 참여 부탁드립니다 :) 🙋‍♂️🙋‍♀️

퀴즈 정답자 분들께
치킨과 커피 쿠폰이 준비되어 있어요!

역시~ 금요일 저녁은 치맥이지! 🍗

DATADOG

DORA란 무엇인가요?

Dev**O**ps **R**esearch and **A**ssessment

- DevOps의 성숙도와 성과를 연구하고 평가하는 조직

What do they do?

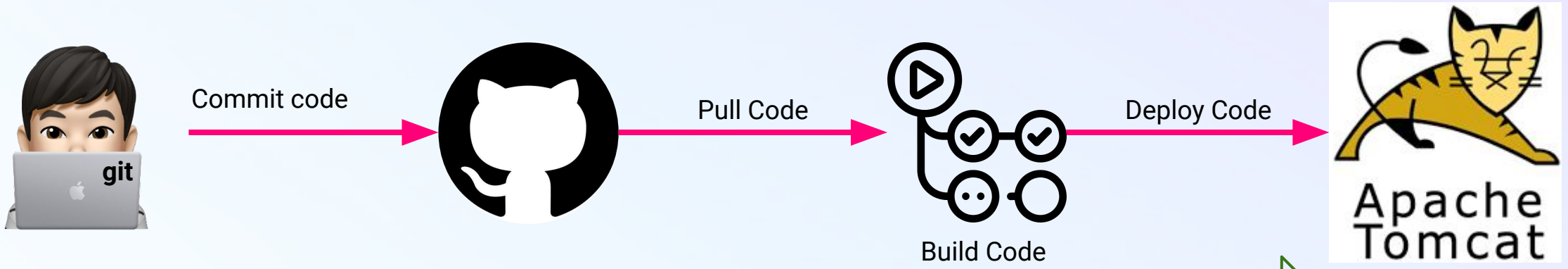
- State of DevOps 보고서 발행

- DevOps 성숙도 모델 제공

Performance level	Change lead time	Deployment frequency	Change fail rate	Failed deployment recovery time	Percentage of respondents*
Elite	Less than one day	On demand (multiple deploys per day)	5%	Less than one hour	19% (18-20%)
High	Between one day and one week	Between once per day and once per week	20%	Less than one day	22% (21-23%)
Medium	Between one week and one month	Between once per week and once per month	10%	Less than one day	35% (33-36%)
Low	Between one month and six months	Between once per month and once every six months	40%	Between one week and one month	25% (23-26%)

1. DORA Metrics (변경 리드 타임)

DEVELOP | **Build + Test + Deploy** | **Production**



1. Lead Time for Changes (Code Commit 부터 배포까지 시간)

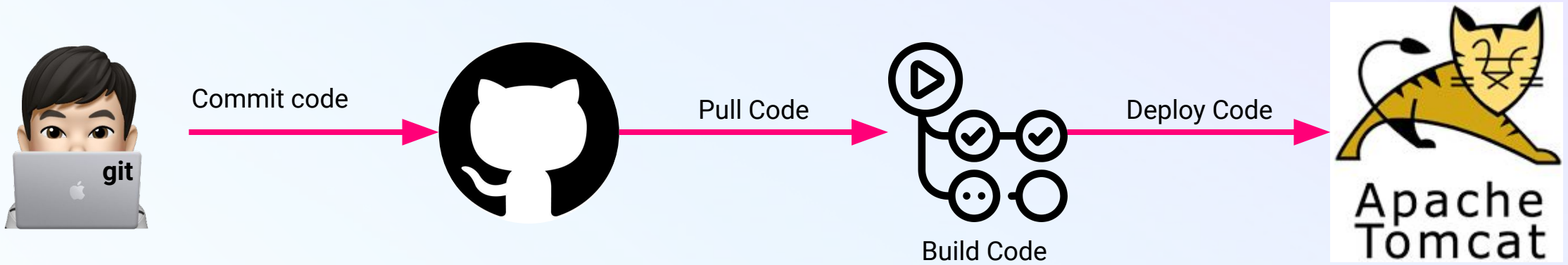
1) Lead Time for Changes

- Commit 후 프로덕션으로 배포되기까지의 시간

Elite	High팀	Medium
하루 이내 반영	한 주 이내 반영	한 달 이내 반영

2. DORA Metrics (배포 빈도)

DEVELOP | **Build + Test + Deploy** | **Production**



2. 배포 빈도

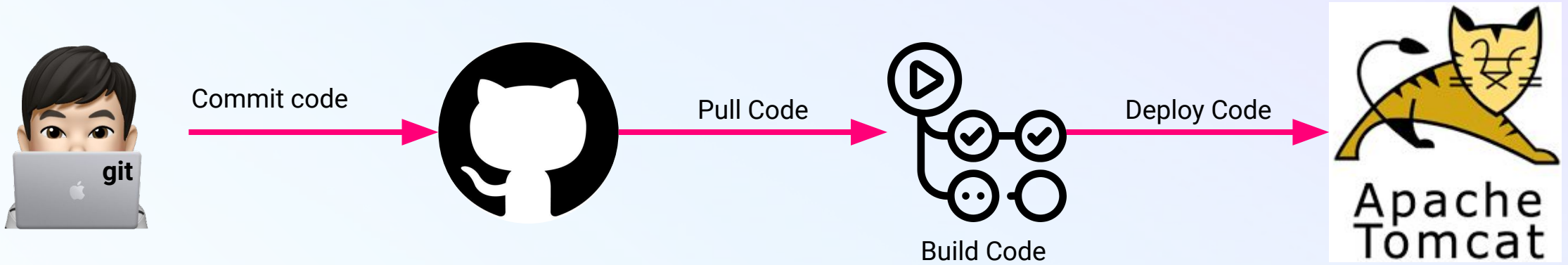
2) Deployment Frequency

- 배포 빈도

Elite	High	Medium
하루에 두번 이상	주 1회 이상	월 1회 이상

3. DORA Metrics (변경 실패율)

DEVELOP | **Build + Test + Deploy** | **Production**



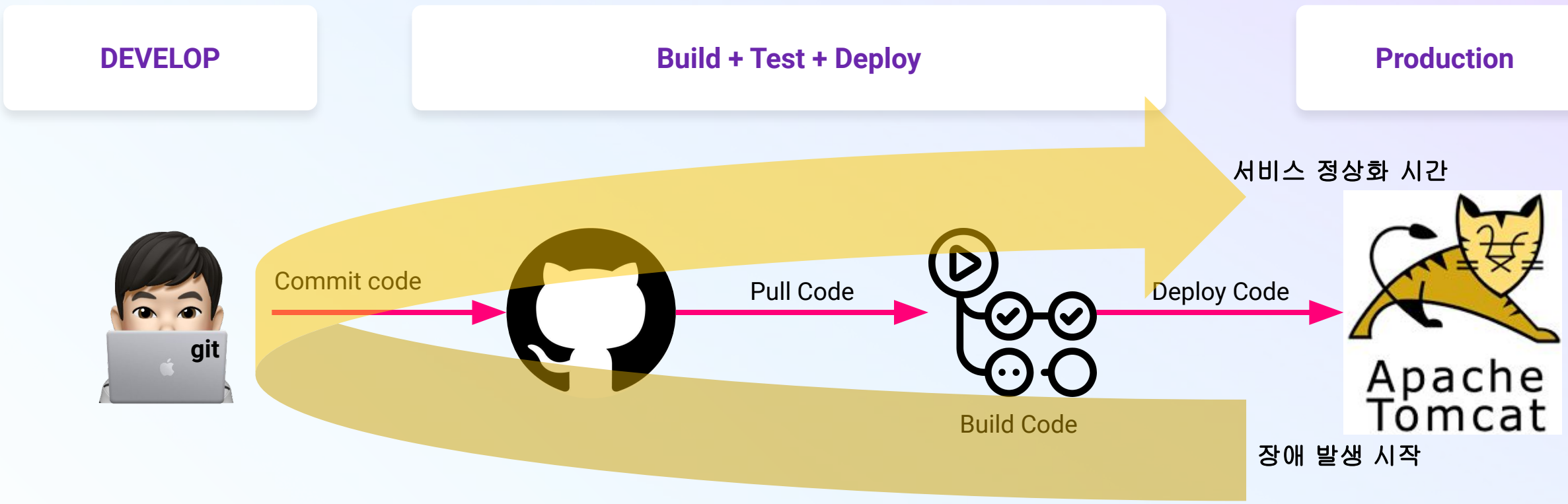
3. 변경 실패율

3) Change Failure Rate

- Production 배포 후 장애 발생 비율

Elite	High	Medium
5% 이내	20% 이내	10% 이내

4. DORA Metrics (장애 복구 시간)



4) Mean Time to Recovery

- 장애 발생 후 복구까지 얼마나 걸리는지?

Elite	High	Medium
한 시간 미만	하루 미만	하루 미만

Elite Performance 팀의 비즈니스 영향도

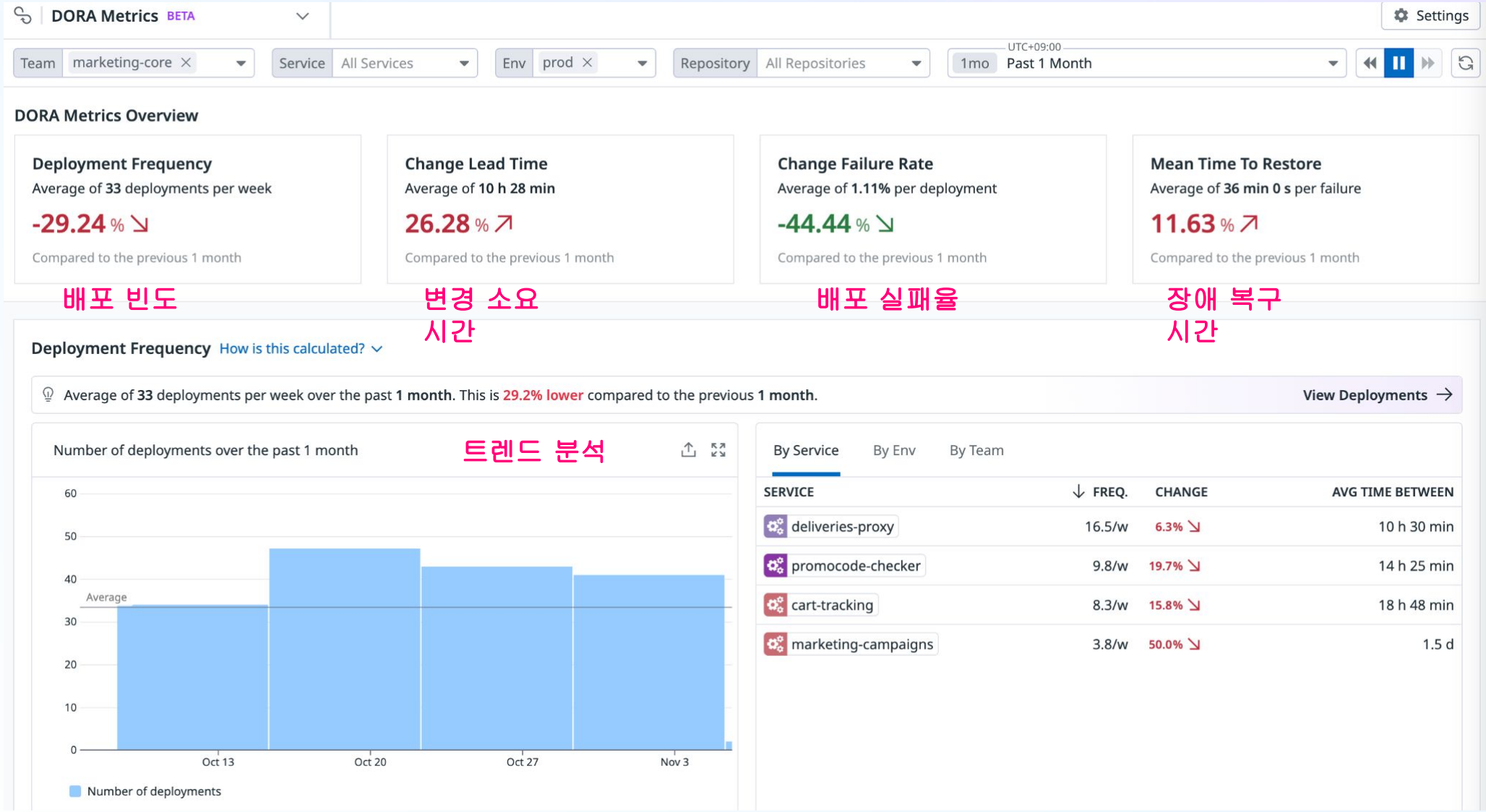
Performance level	Change lead time	Deployment frequency	Change fail rate	Failed deployment recovery time	Percentage of respondents*
Elite	Less than one day	On demand (multiple deploys per day)	5%	Less than one hour	19% (18-20%)

- 고객 요구 사항에 신속히 대응 => 사용자 경험 향상
- 불필요한 버그 수정을 줄여 핵심 기능에 더 많은 시간 투자 => 직원 업무 만족도 증가

엘리트팀과 같은 성과를 내는것 보다 개선을 시도하는것이 더욱 중요

⇒ 개선을 위한 첫 스텝은 현재 수준을 측정하고 개선 포인트를

DORA 메트릭을 활용한 대시보드 샘플



1. 배포 빈도와 배포 리드 타임

배포 빈도 트래킹 방법?

DEVELOP

Build + Test + Deploy

Production

Commit code

Pull Code

Build Code

Deploy Code

Apache Tomcat

APM(or API)

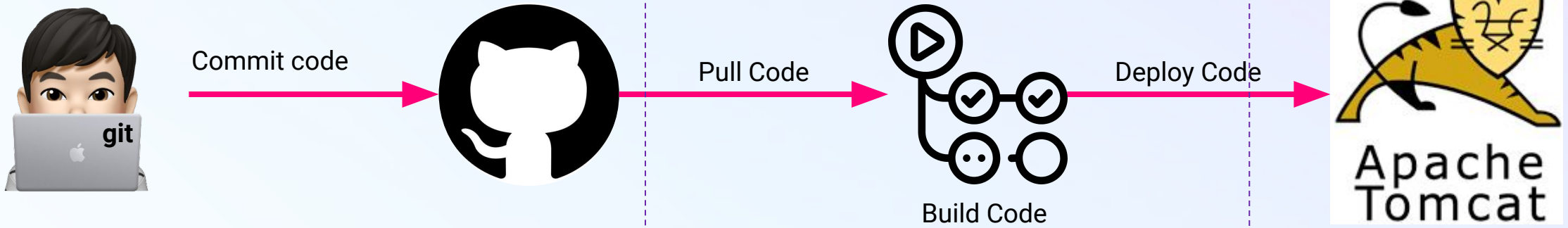
DORA Metrics

APM에서 버전(DD_VERSION) 변경을 감지하여 배포 빈도 카운트

dora.deployments.count | dora.change_lead_time | dora.incidents.count | dora.time_to_restore

변경 리드 타임 트래킹 방법?

DEVELOP | Build + Test + Deploy | Production



1. Github 연동

2. APM(or API)

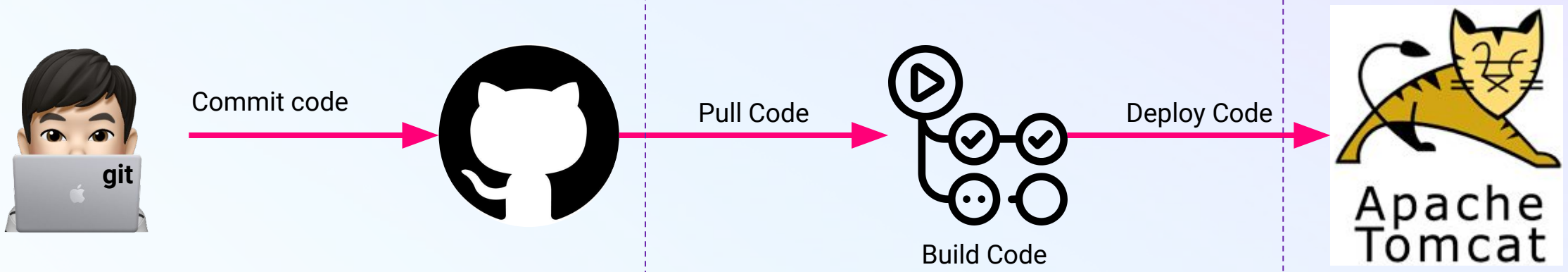
DORA Metrics

Commit SHA 값을 기준으로 커밋한 시간 ==> 배포된 시간을 계산

dora.deployments.count	dora.change_lead_time	dora.incidents.count	dora.time_to_restore

변경 리드 타임 트래킹 방법?

DEVELOP | **Build + Test + Deploy** | **Production**



Github

EX) 어제 14시에 Commit 후 오늘 15시에 Production 배포가 되었다

Commit SHA 값을 기준으로 커밋한 시간 ==> 배포된 시간을 계산

dora.deployments.count	dora.change_lead_time	dora.incidents.count	dora.time_to_restore

연동 완료 후 배포 빈도가 어떻게 보이나요?

DORA Metrics Overview

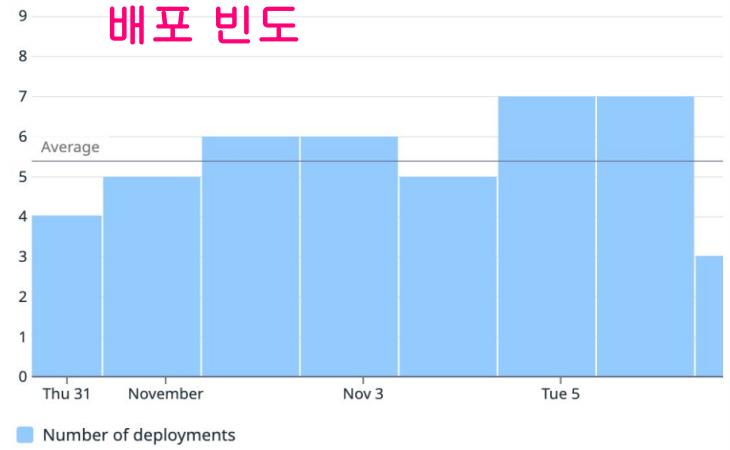
Deployment Frequency
Average of **5** deployments per day
-14 % ↘
Compared to the previous 1 week

Change Lead Time
Average of **7 h 10 min**
-1.8 % ↘
Compared to the previous 1 week

Deployment Frequency How is this calculated?

Average of **5** deployments per day over the past **1 week**. This is **14.0% lower** com

Number of deployments over the past 1 week



Deployments Breakdown

Open in a New Tab

Team All Teams | Service All Services | Env prod | Repository All Repositories

UTC+09:00 1w Past 1 Week

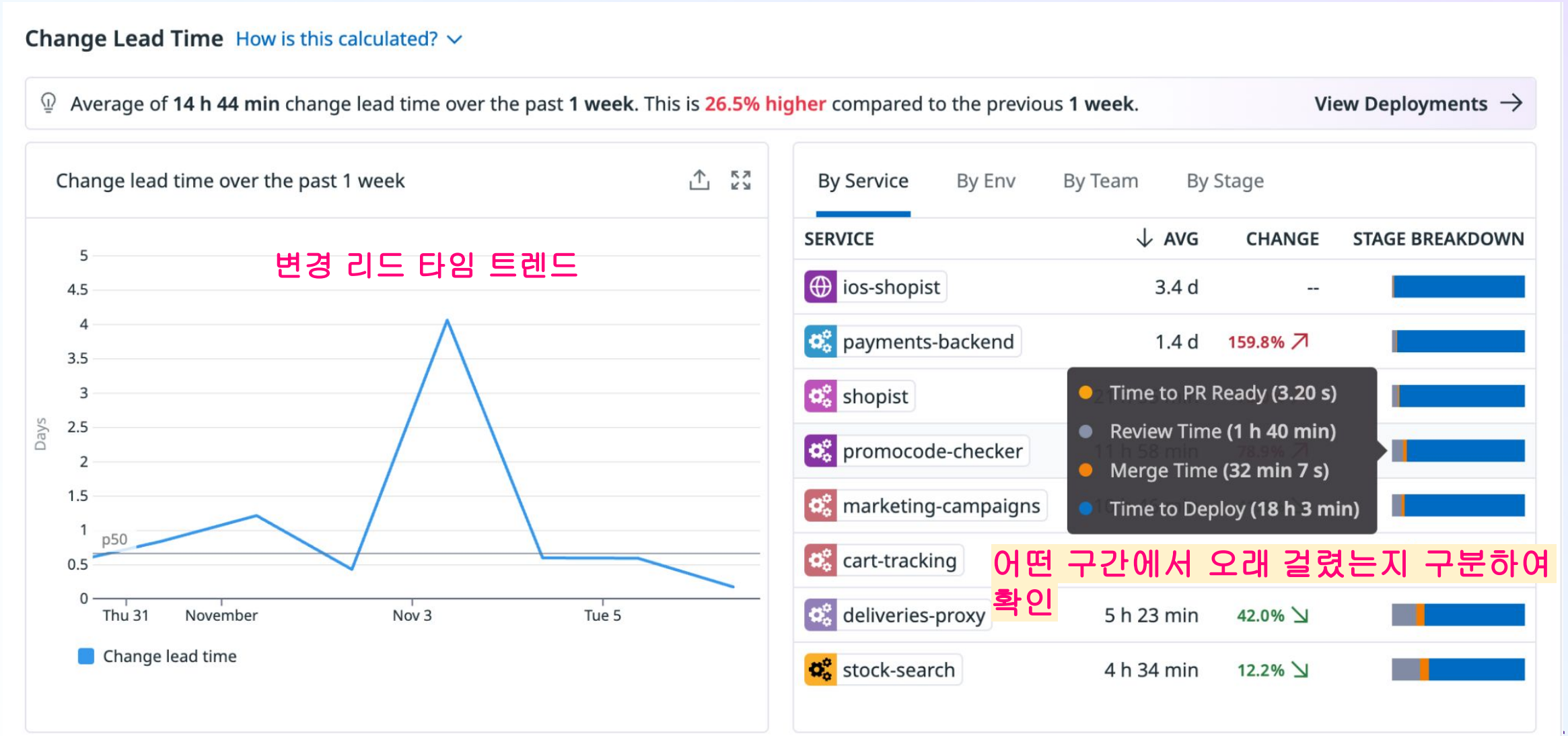
Deployment Frequency
Average of **11** deployments per day over the past **1 week**
-2.27 % ↘
Compared to the previous 1 week

Change Lead Time
Average of **14 h 44 min** change lead time over the past **1 week**
26.45 % ↗
Compared to the previous 1 week

TIME	VERSION	ENV	SERVICE	DURATION	LEAD TIME	BREAKDOWN
Nov 6, 1:39:45 ...	3dd7380	prod	competitor-analysis	1 h 43 min		
Nov 6, 1:39:44 ...	3dd7380	prod	stock-search	2 h 41 min		
Nov 6, 9:51:12 ...	29b9cce	prod	promocode-checker	1 h 44 min	3 h 49 min	
Nov 6, 9:50:48 ...	648e26c	prod	stock-search	3 h 59 min	3 h 48 min	
Nov 6, 1:40:59 ...	c1fe01f	prod	promocode-checker	1 h 11 min	9 h 39 min	
Nov 6, 1:40:59 ...	c1fe01f	prod	deliveries-proxy	3 h 1 min	4 h 39 min	
Nov 6, 1:40:59 ...	c1fe01f	prod	cart-tracking	4 h 49 min	10 h 38 min	
Nov 6, 1:40:59 ...	c1fe01f	prod	competitor-analysis	1 h 13 min		

배포 히스토리

연동 완료 후 변경 리드 타임이 어떻게 보이나요?

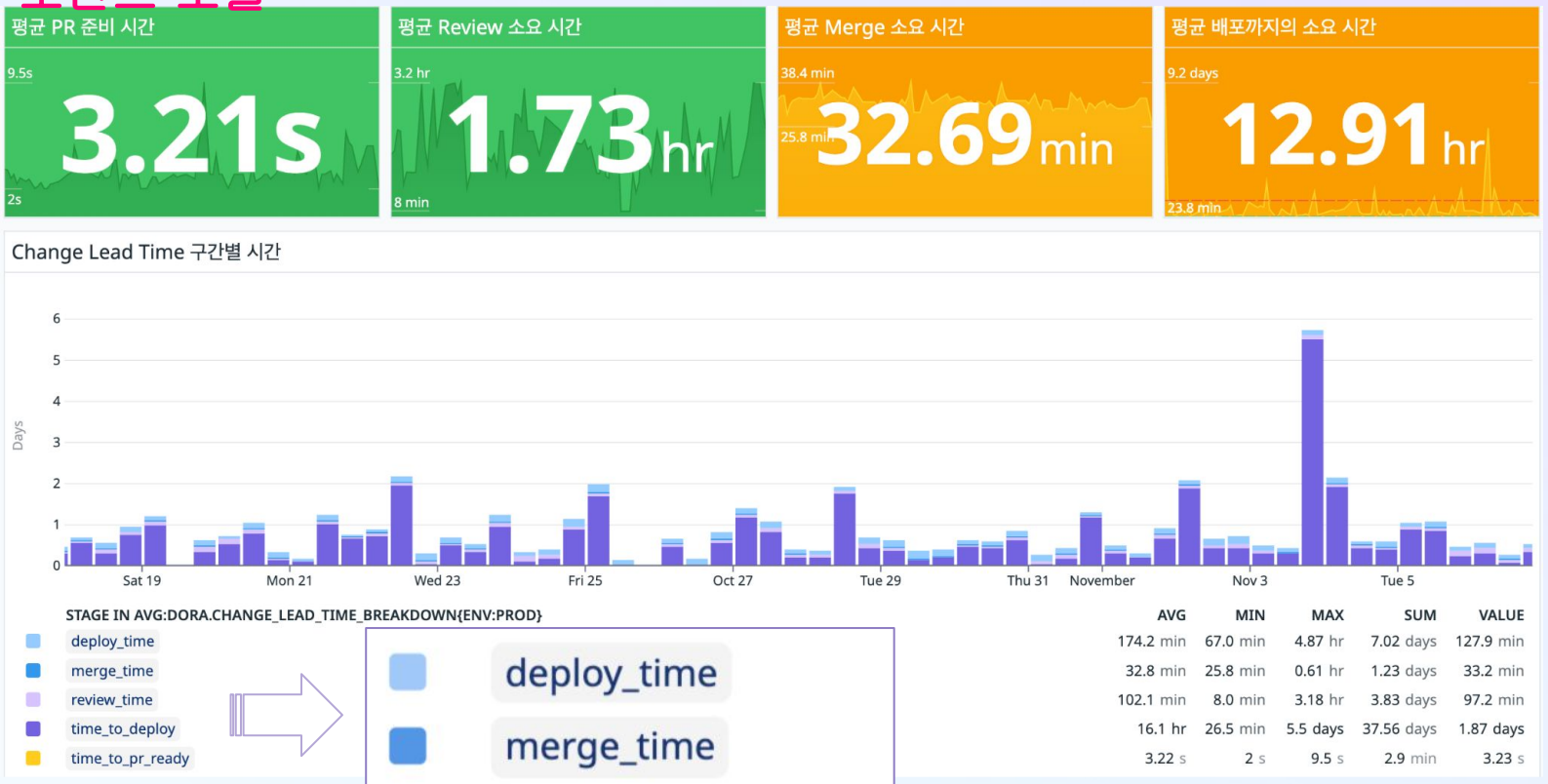


변경 리드 타임 개선하기

1. **메트릭 분석을 통한 병목 확인**
   -> dora.change_lead_time_breakdown

2. CI 파이프 라인 병목 분석
   -> CI Visibility

PR, Review, Merge, Deploy 등 단계별 소요시간을 분석하여 개선 포인트 도출

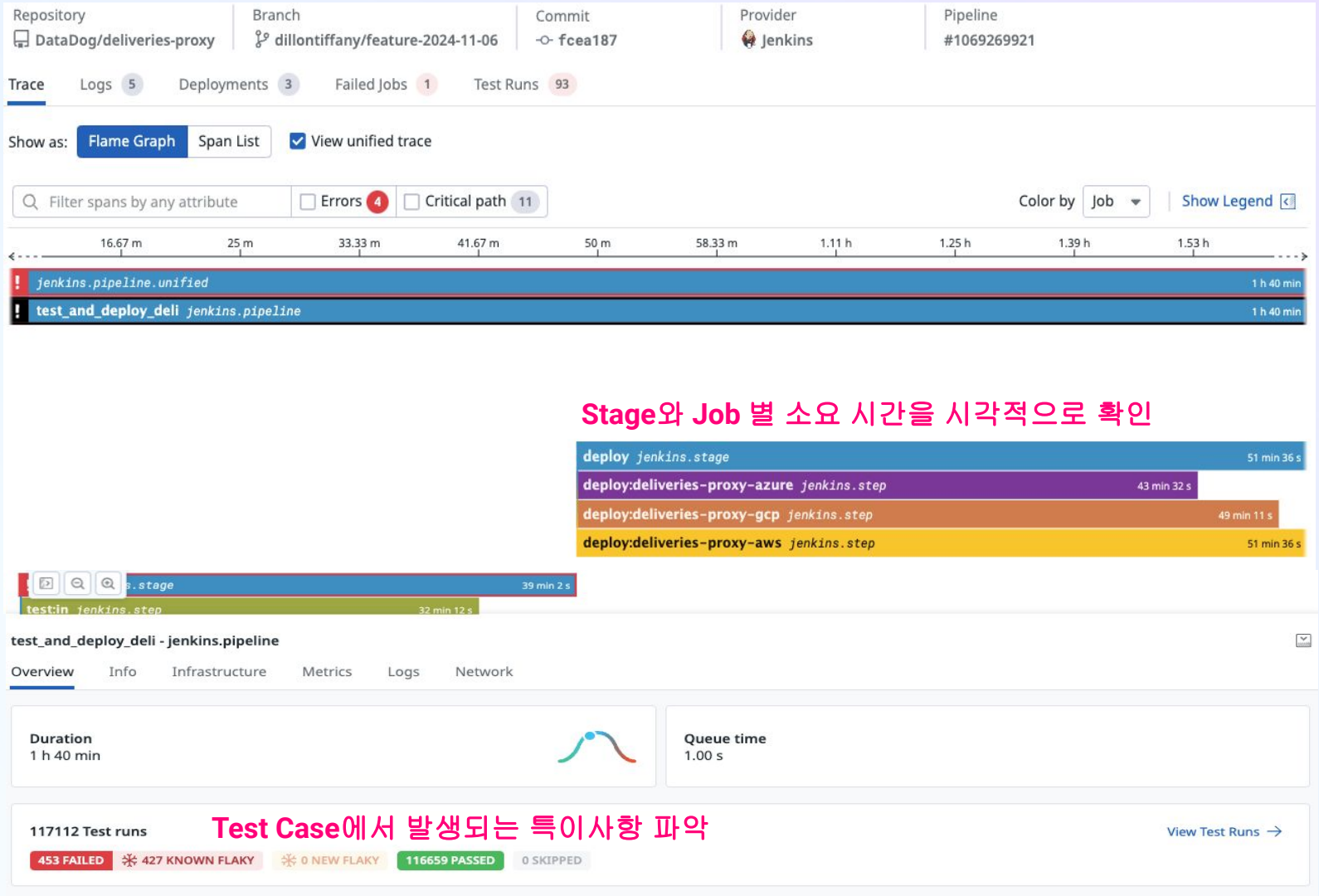


변경 리드 타임 개선하기

1. 메트릭 분석을 통한 병목 확인
   -> dora.change_lead_time_breakdown

2. **CI 파이프 라인 병목 분석**
   -> CI Visibility

CI 파이프라인의 Stage, Job 별 소요 시간을 분석하여 개선 포인트 도출



QUIZ 🤔

변경 리드 타임을 개선 하려고 합니다.
CI 파이프 라인의 병목을 분석하기 위해 사용하면 좋은 솔루션은?

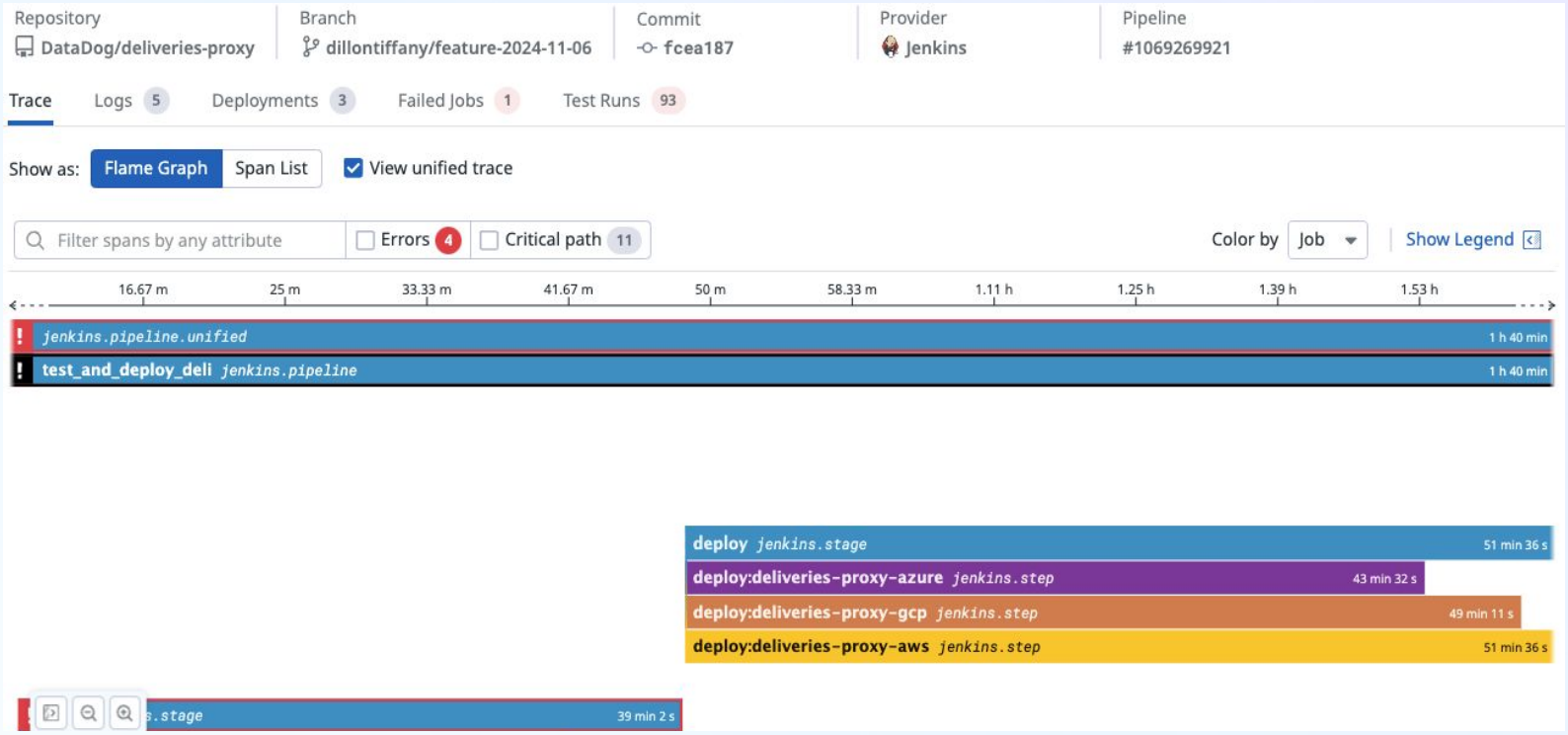
1. Software Composition Analysis
2. CSI 수사대
3. Test Optimization
4. CI Visibility
5. Quality Gate

QUIZ 🤔

변경 리드 타임을 개선 하려고 합니다.
CI 파이프 라인의 병목을 분석하기 위해 사용하면 좋은 솔루션은?

1. Software Composition Analysis
2. CSI 수사대
3. Test Optimization
4. **CI Visibility**
5. Quality Gate

CI 파이프라인의 Stage, Job 별 소요 시간을 분석하여 개선 포인트 도출



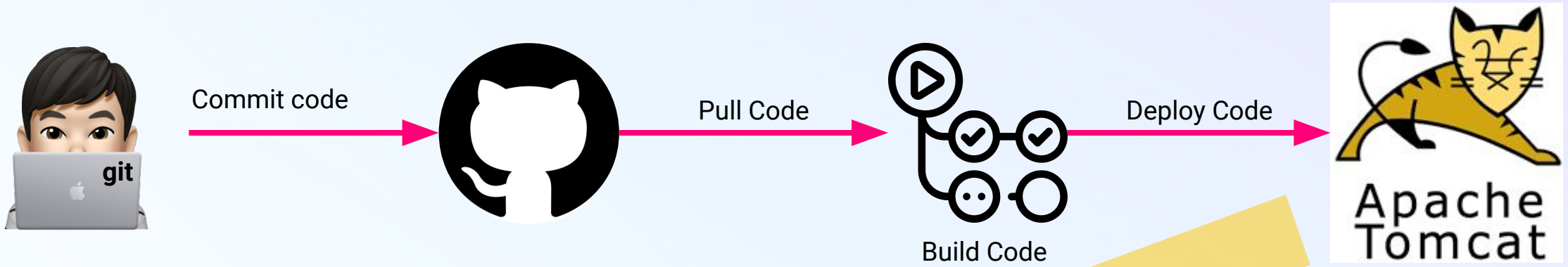
2. 변경 실패율

변경 실패율 트래킹

DEVELOP

Build + Test + Deploy

장애 발생 시작
Production



배포 빈도 수

장애사항 인시던트 발생 수

DORA Metrics

dora.deployments.count | dora.change_lead_time | dora.incidents.count | dora.time_to_restore

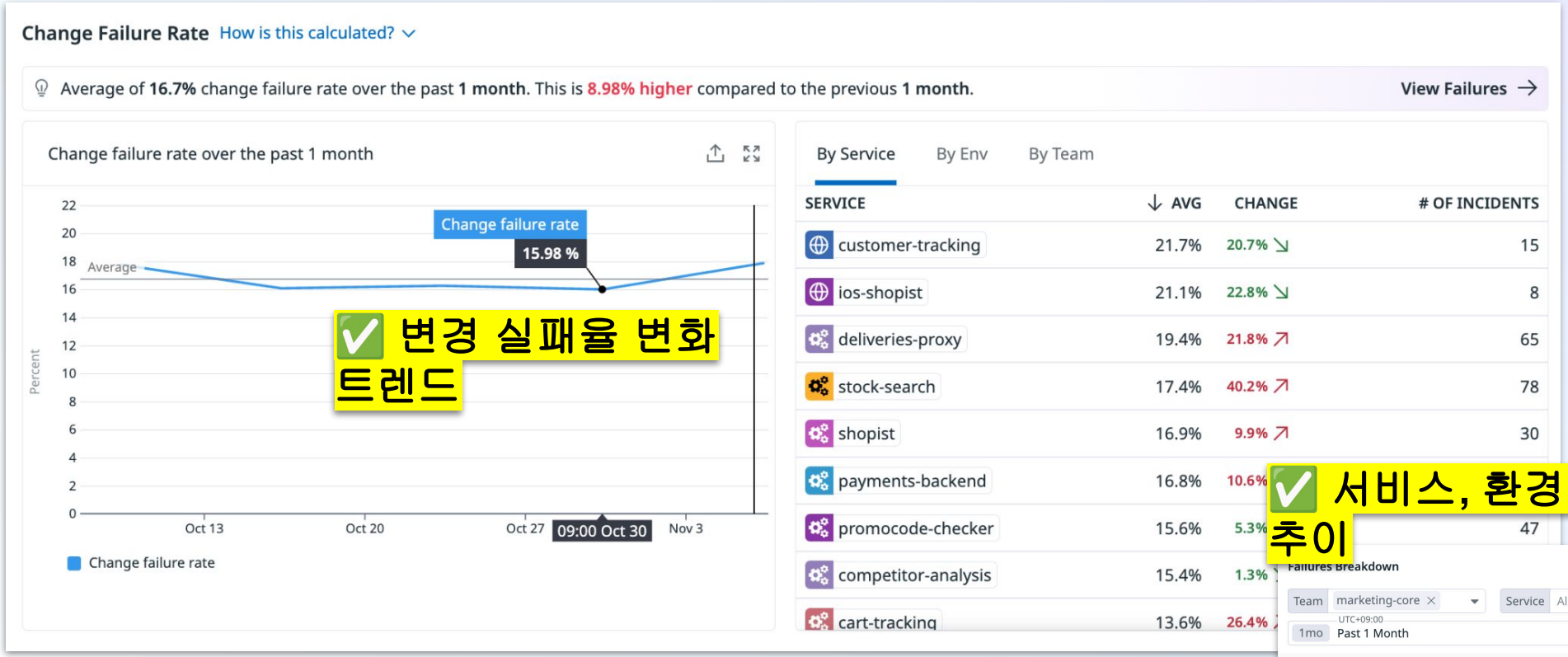
🧐 변경 실패율은 어떻게 측정하나요?

변경 실패율 = $\frac{\text{dora.incidents.count (인시던트 발생 수)}}{\text{dora.deployments.count (배포 수)}}$

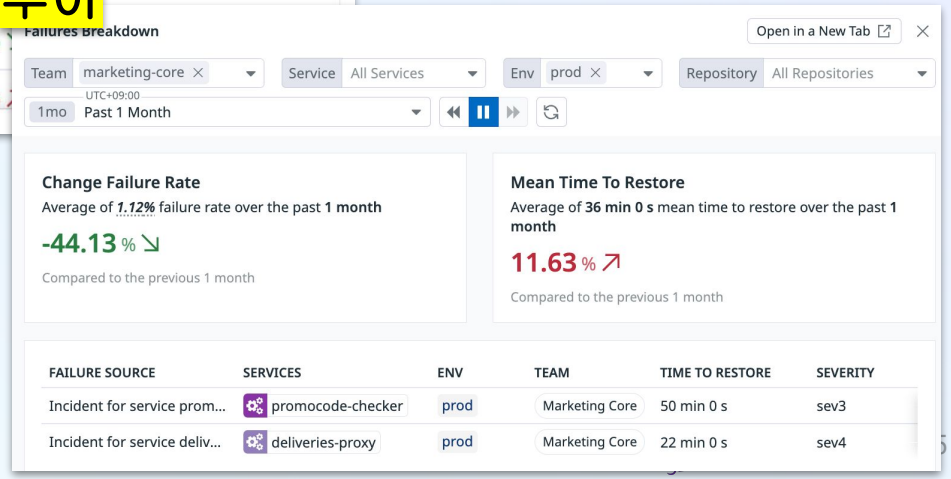
→ 배포시 실패하는 확률


```
{
 "data": {
   "attributes": {
     "services": [
       "notes"
     ],
     "team": "Backend-team",
     "started_at": 1730710775629,
     "finished_at": 1730711075629,
     "git": {
       "commit_sha":
"b97de359da9ee657667d4b86fbeb71d35a21463e",
       "repository_url":
"https://github.com/dd-dog/code-analysis.git"
     },
     "env": "dev",
     "name": "notes 이슈 발생 긴급!",
     "severity": "SEV-1"
   }
 }
}
```


변경 실패율은 Datadog에서는 어떻게 보이나요? 📈



✅ 서비스, 환경, 팀별 변경 실패율 추이

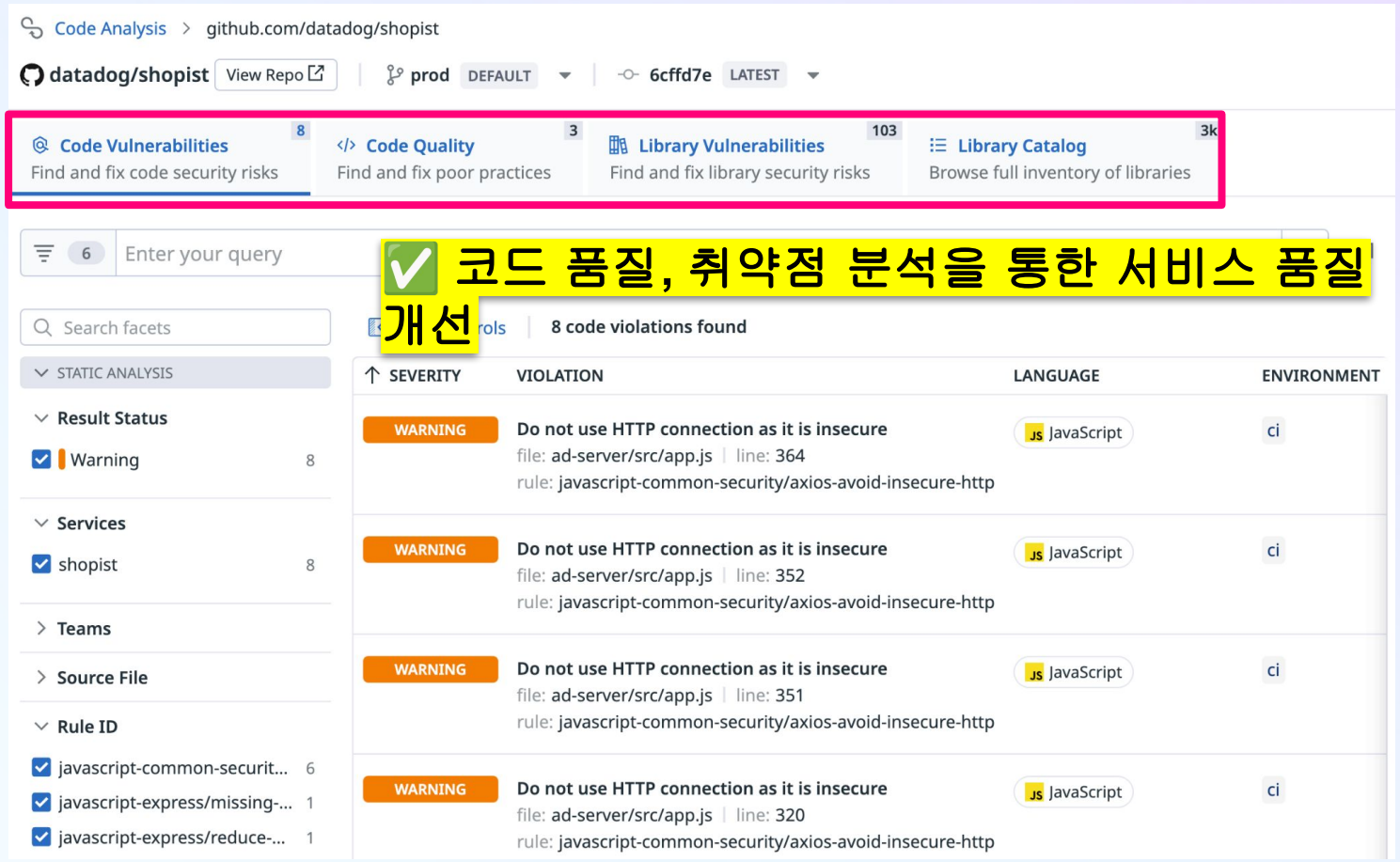


변경 실패율을 개선하기 위한 방법 🍯 (Tip) 💡

1️⃣ 코드 퀄리티 개선
-> Code Analysis, Quality Gate

2️⃣ 테스트 특이사항 탐지
-> Test Optimization

3️⃣ 카나리 환경에서 신속한 이슈 탐지
-> APM Deployment Tracking

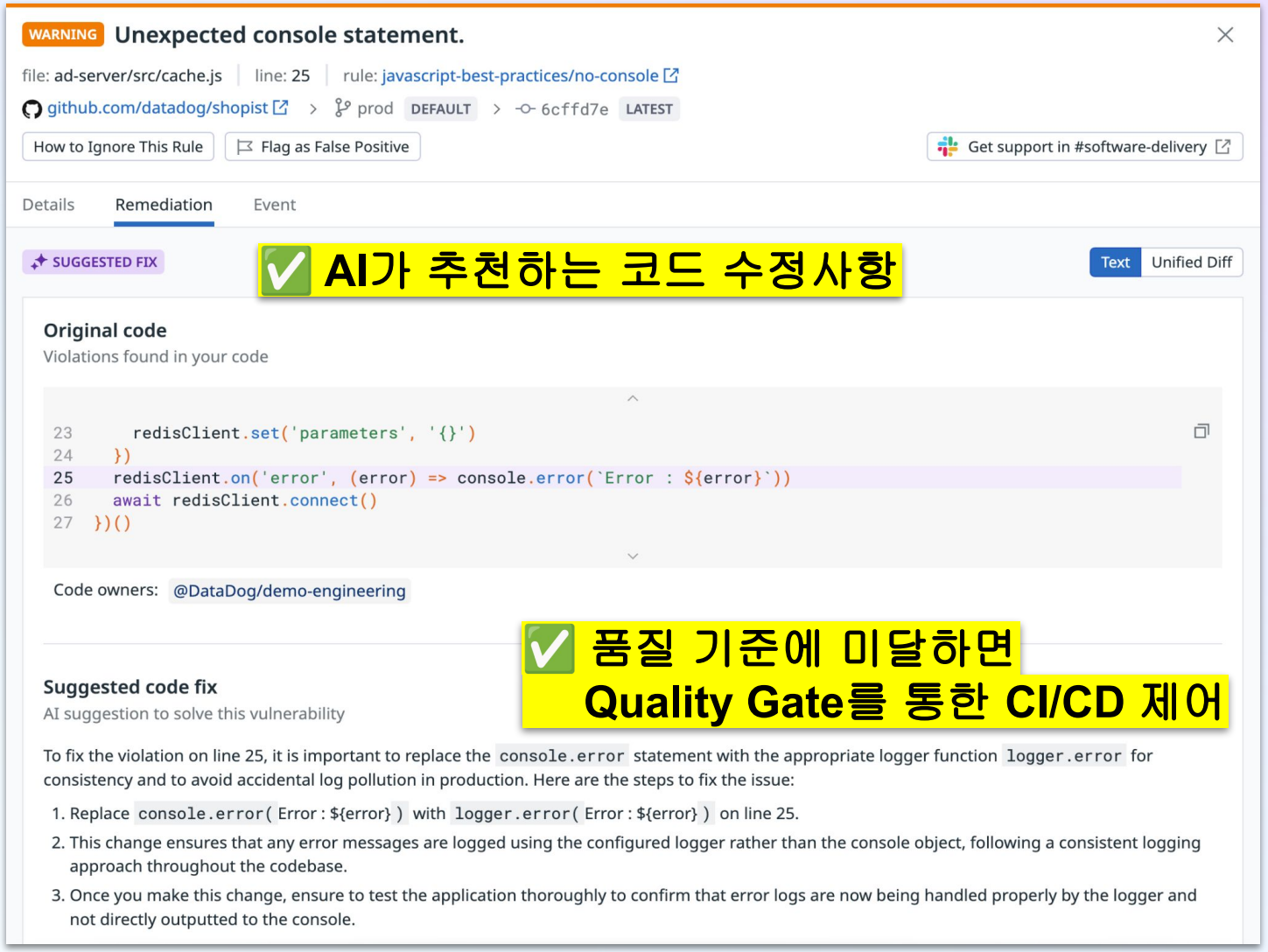


변경 실패율을 개선하기 위한 방법 🍯 (Tip) 💡

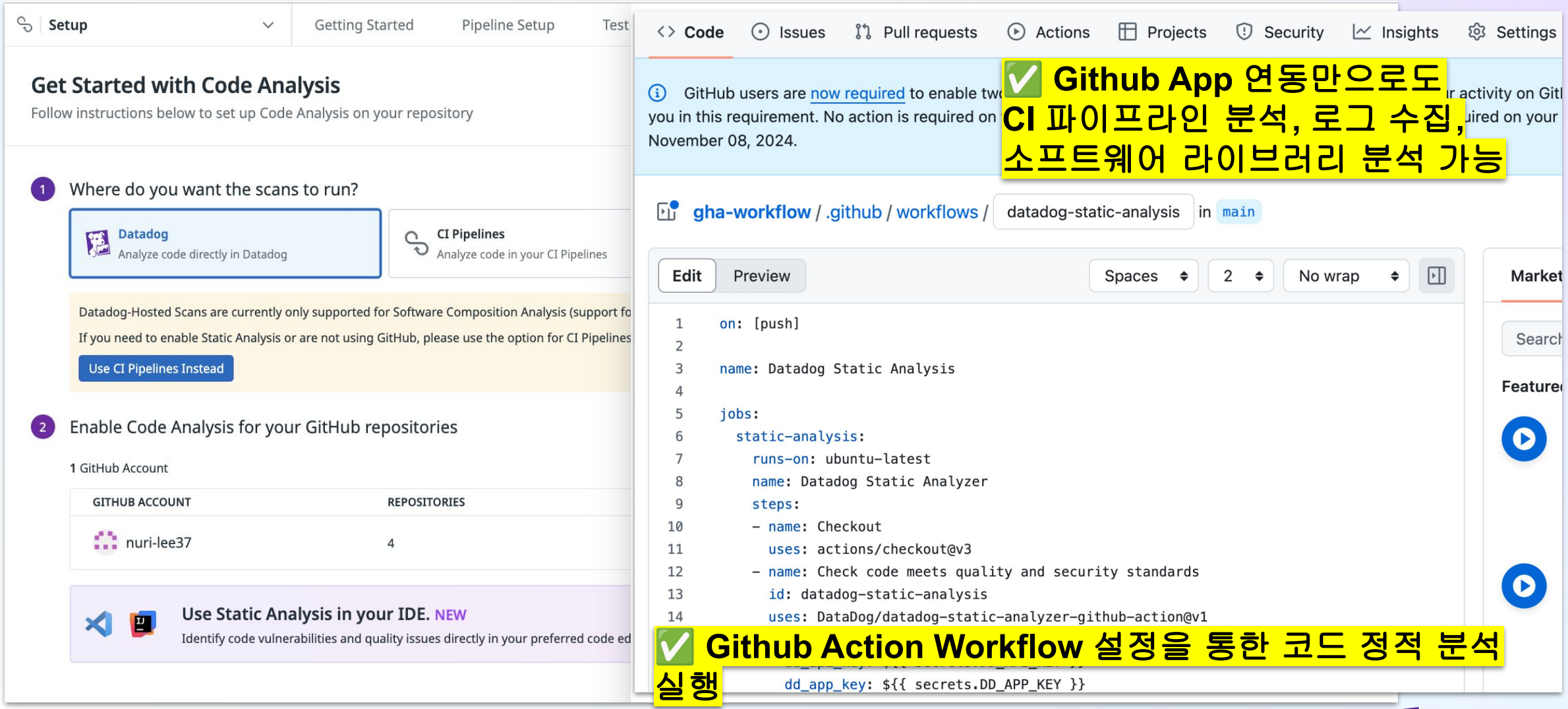
1️⃣ 코드 퀄리티 개선
-> Code Analysis, Quality Gate

2️⃣ 테스트 특이사항 탐지
-> Test Optimization

3️⃣ 카나리 환경에서 신속한 이슈 탐지
-> APM Deployment Tracking



정적 분석, 소프트웨어 라이브러리 분석은 어떻게 설정하나요?

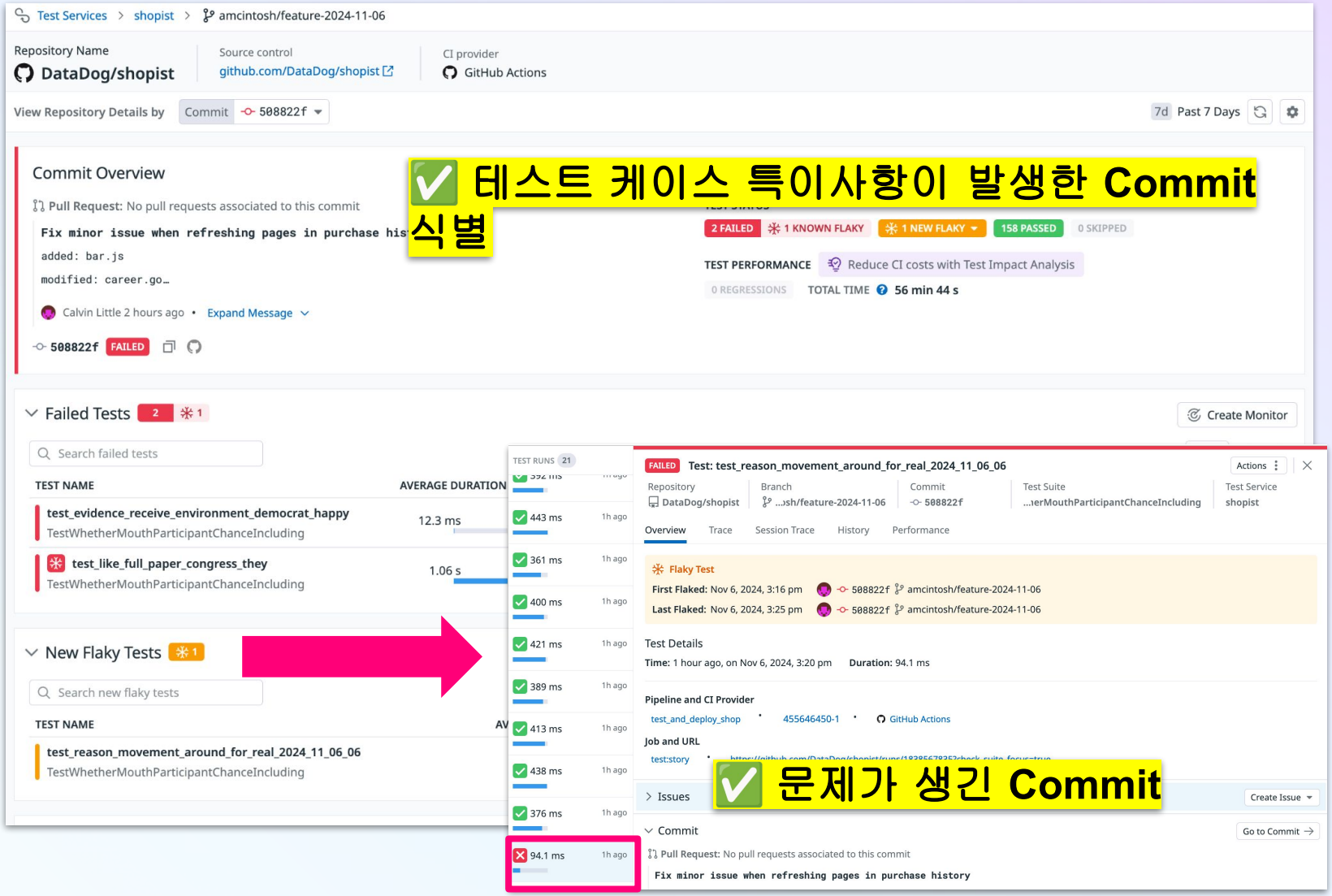


변경 실패율을 개선하기 위한 방법 🍯 (Tip) 💡

1️⃣ 코드 퀄리티 개선
-> Code Analysis, Quality Gate

2️⃣ **테스트 특이사항 탐지**
-> **Test Optimization**

3️⃣ 카나리 환경에서 신속한 이슈 탐지
-> APM Deployment Tracking

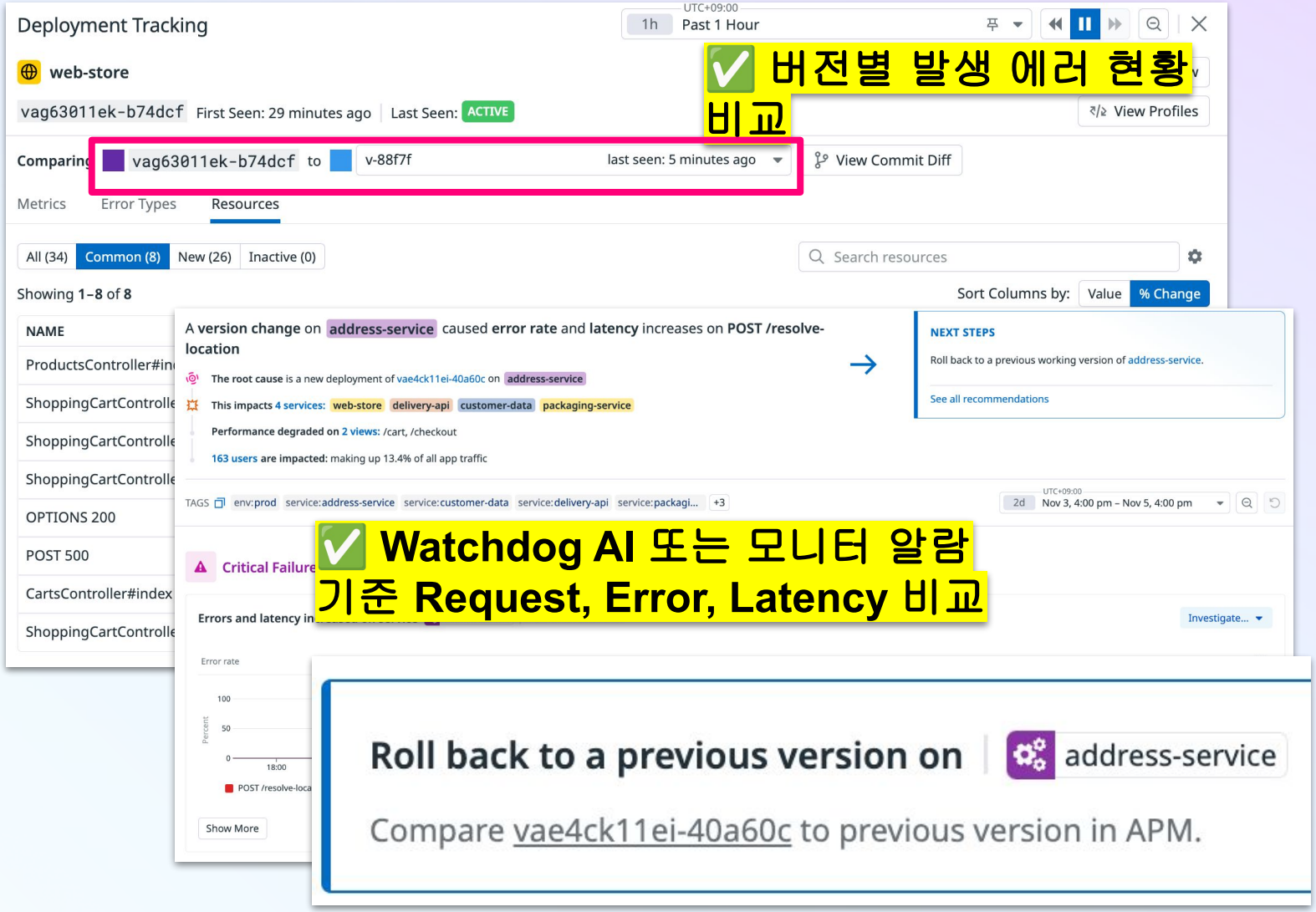


변경 실패율을 개선하기 위한 방법 🍯 (Tip) 💡

1️⃣ 코드 퀄리티 개선
-> Code Analysis, Quality Gate

2️⃣ 테스트 특이사항 탐지
-> Test Optimization

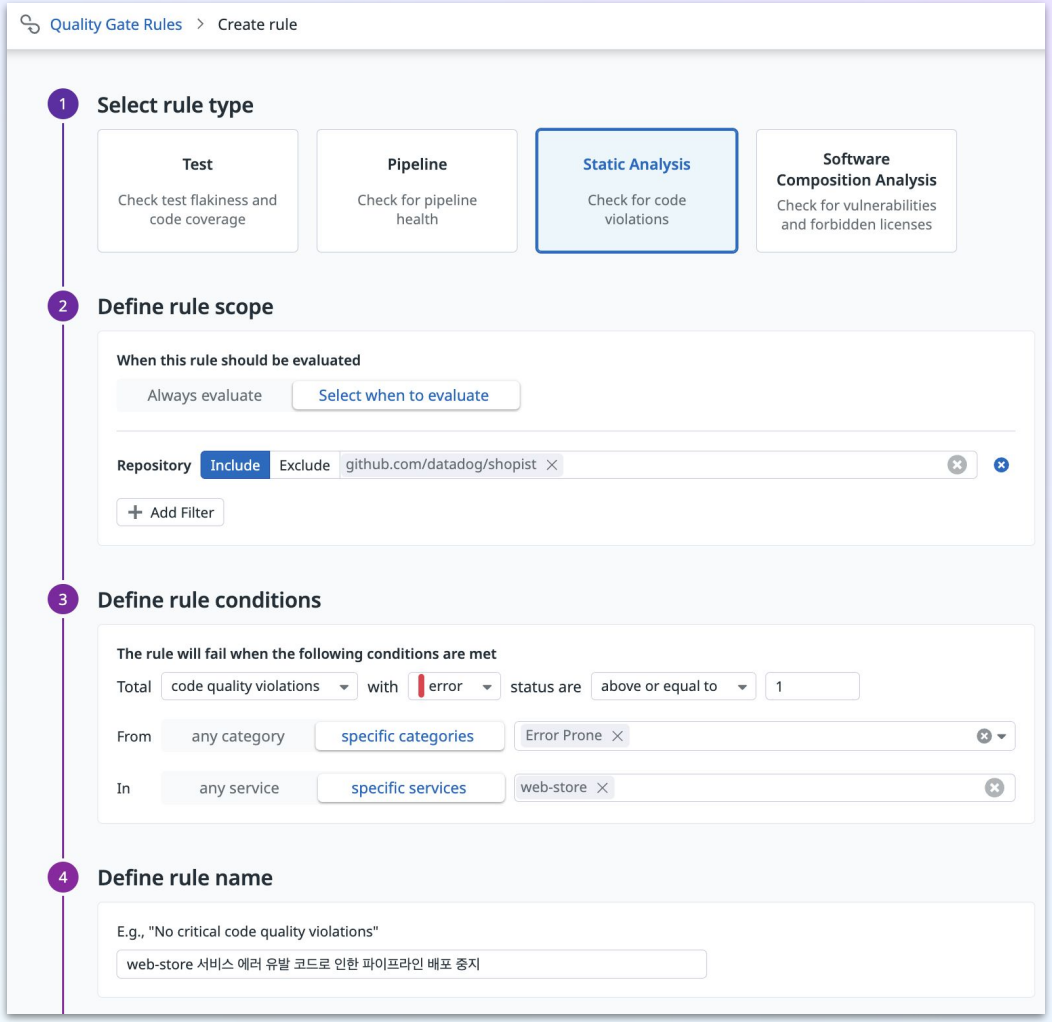
3️⃣ **카나리 환경에서 신속한 이슈 탐지**
-> **APM Deployment Tracking**



QUIZ 🤔

Q. 정적분석 룰 위반 시 파이프라인 실행을 중단시킬 수 있는 것은?

1. CI Visibility
2. Software Composition Analysis
3. Test Optimization
4. Software Gate
5. Quality Gate

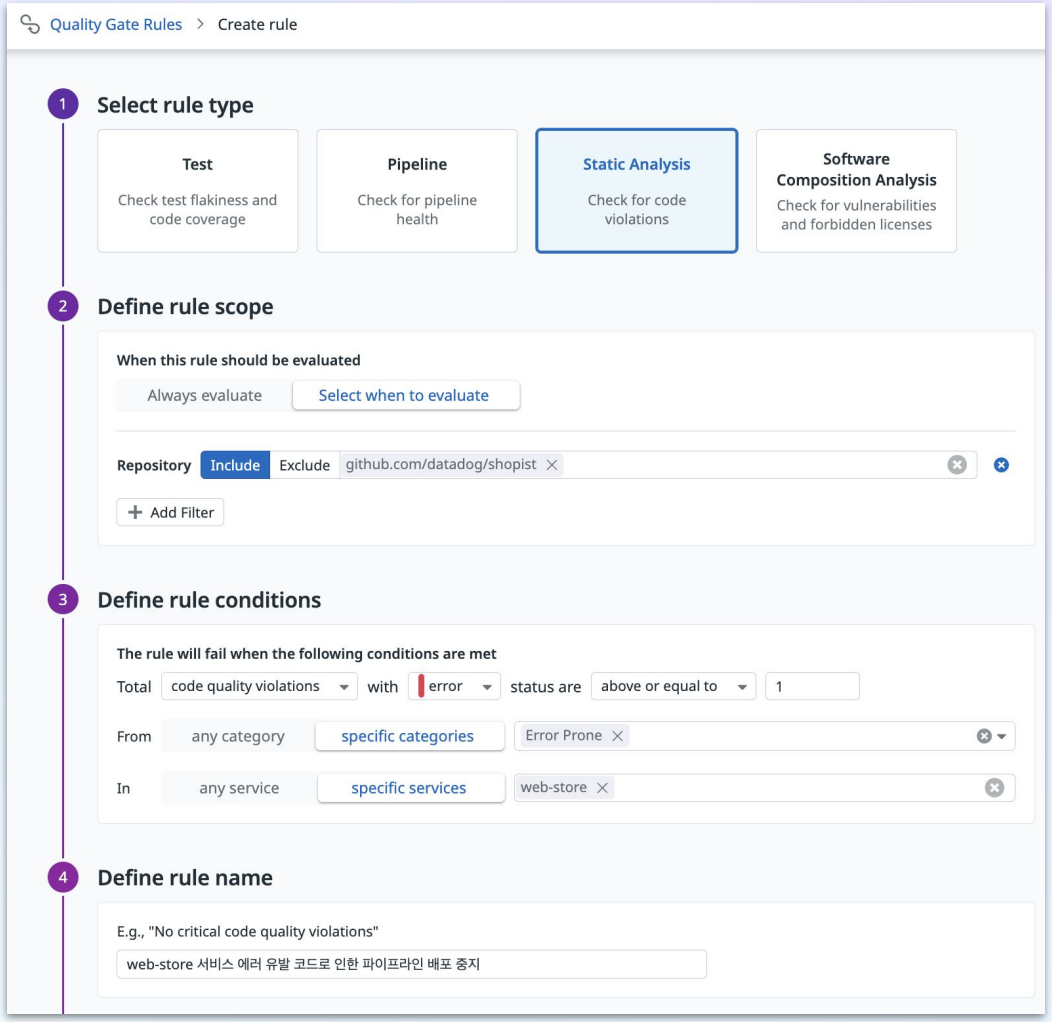


DATADOG

QUIZ 🤔

Q. 정적분석 룰 위반 시 파이프라인 실행을 중단시킬 수 있는 것은?

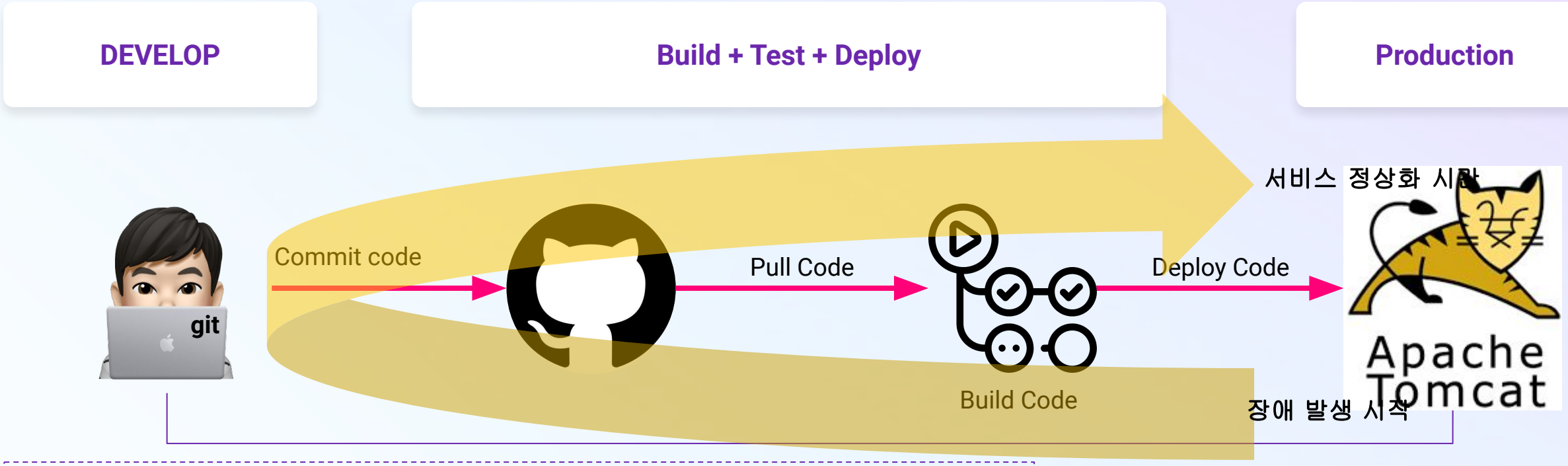
1. CI Visibility
2. Software Composition Analysis
3. Test Optimization
4. Software Gate
5. **Quality Gate**



DATADOG

3. 장애 복구 시간

장애 복구 시간 트래킹?

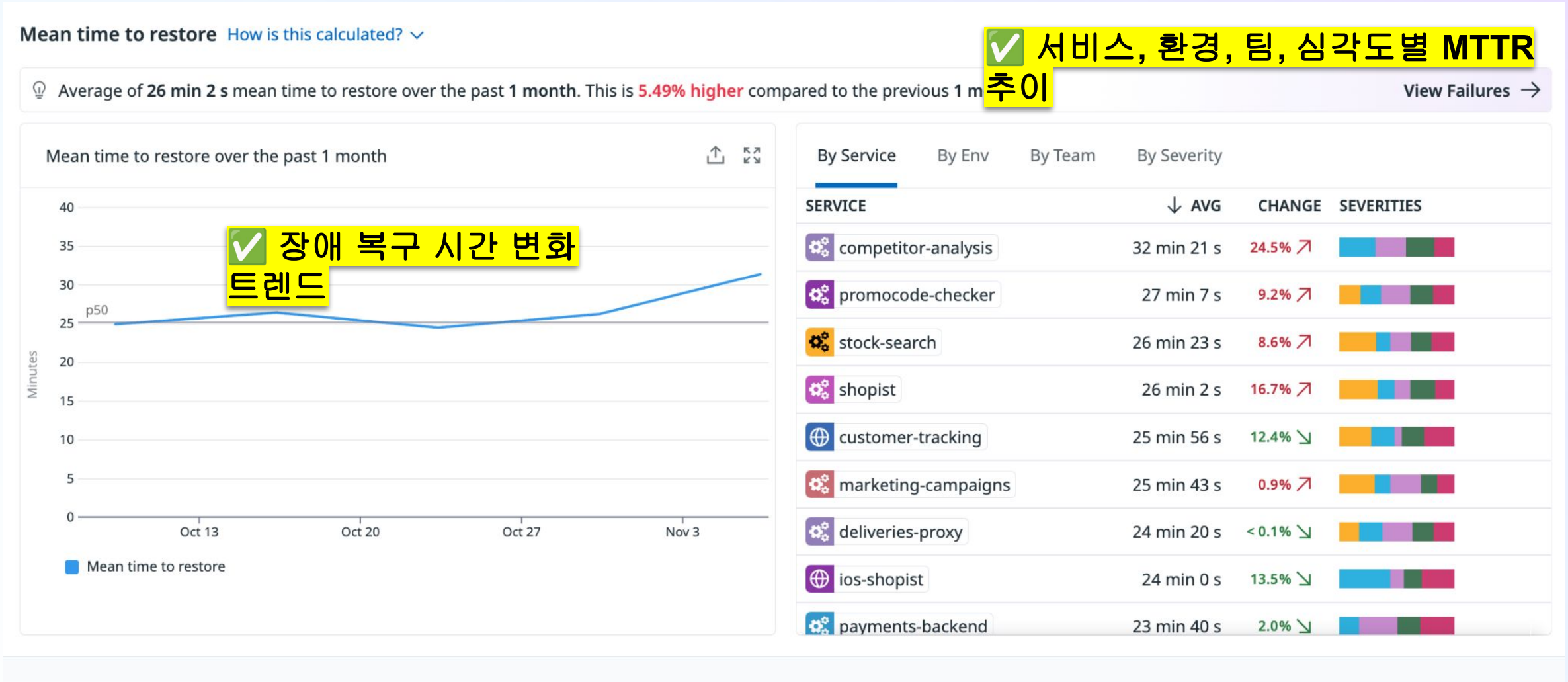


서비스 정상화 시간 - 장애 발생 시작 => API 로 전송

DORA Metrics

dora.deployments.count	dora.change_lead_time	dora.incidents.count	dora.time_to_restore

장애 복구시간은 Datadog에서 어떻게 보이나요? 📈

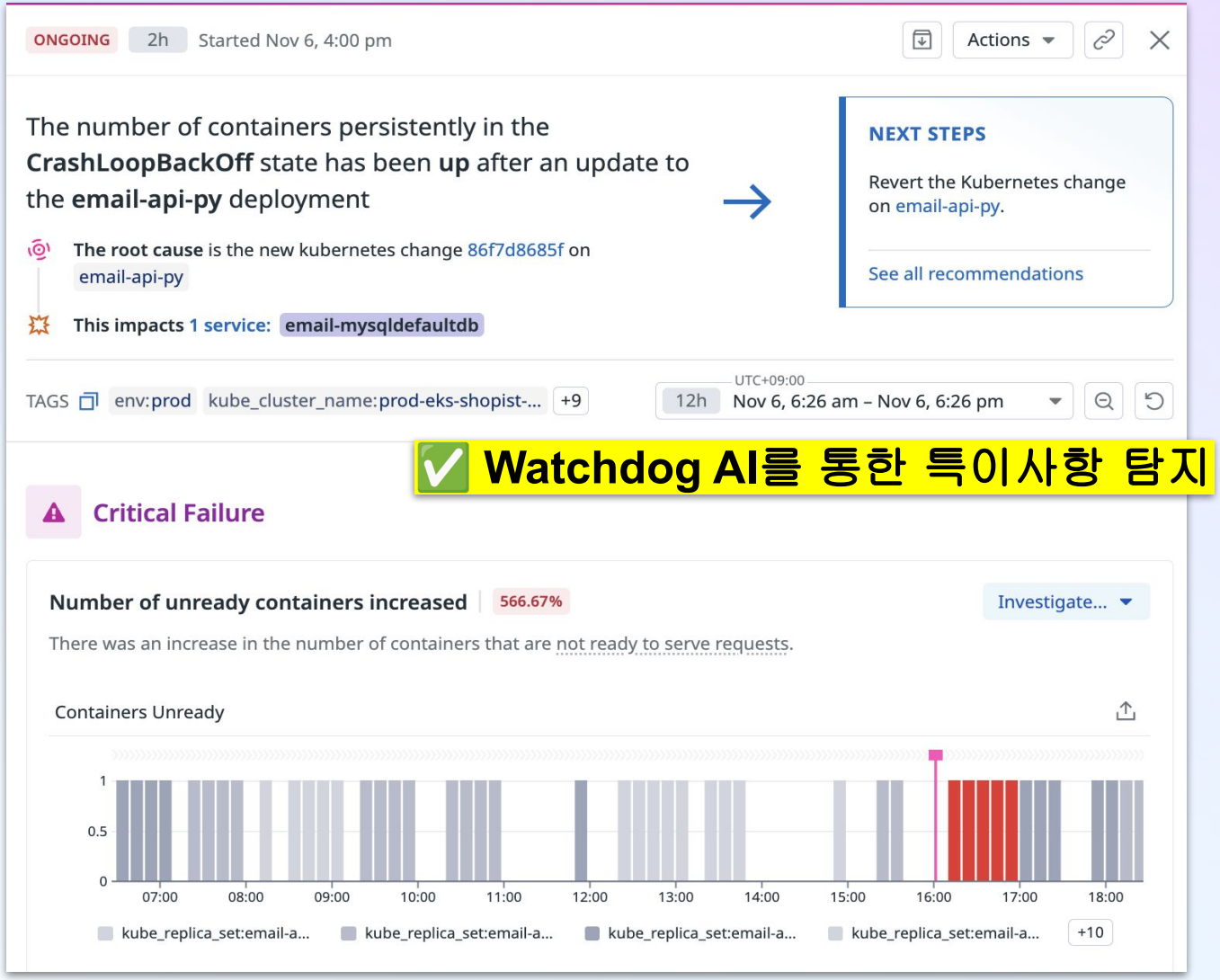


장애 복구 시간을 개선하기 위한 방법 🍯 (Tip) 💡

1️⃣ **빠르게 인지**
-> Watchdog, 모니터, 대시보드

2️⃣ 빠르게 상황 전파
-> Incident, On-Call

3️⃣ 복구 자동화
-> Workflow Automation

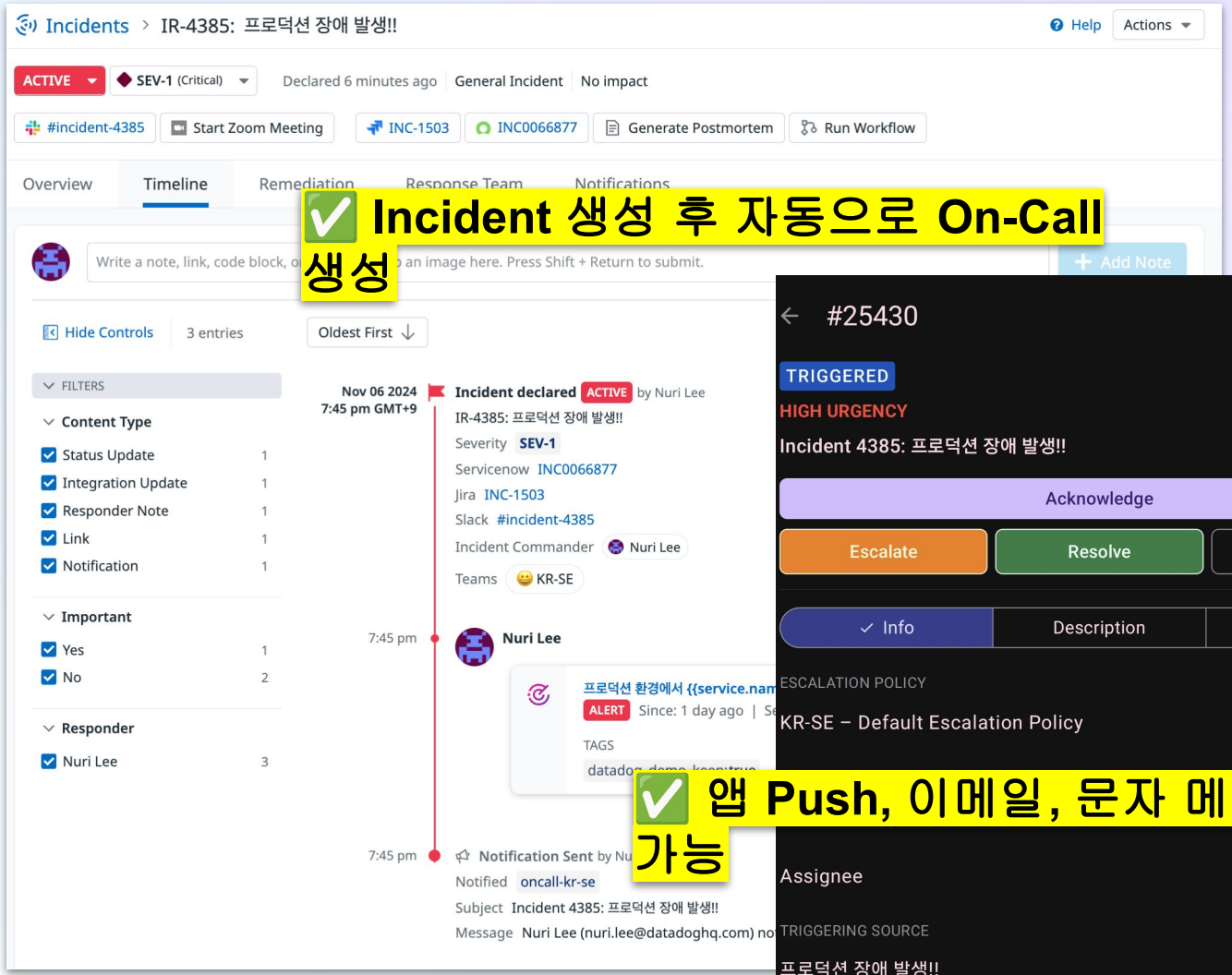


장애 복구 시간을 개선하기 위한 방법 🍯 (Tip) 💡

1️⃣빠르게 인지
-> Watchdog, 모니터, 대시보드

2️⃣**빠르게 상황 전파**
-> Incident Management, On-Call

3️⃣복구 자동화
-> Workflow Automation



장애 복구 시간을 개선하기 위한 방법 🍯 (Tip) 💡

1️⃣빠르게 인지
-> Watchdog, Monitor

2️⃣빠르게 상황 전파
-> Incident, On-Call

3️⃣**복구 자동화**
-> Workflow Automation

✅ 장애 알람 발생 시 자동 롤백 구현

장애 복구 시간을 개선하기 위한 방법 🍯 (Tip) 💡

1️⃣빠르게 인지
-> Watchdog, Monitor

2️⃣빠르게 상황 전파
-> Incident, On-Call

3️⃣**복구 자동화**
-> App Builder

✅ 대시보드를 통해 리소스 제어를 바로 수행하여 신속한 장애 복구 가능

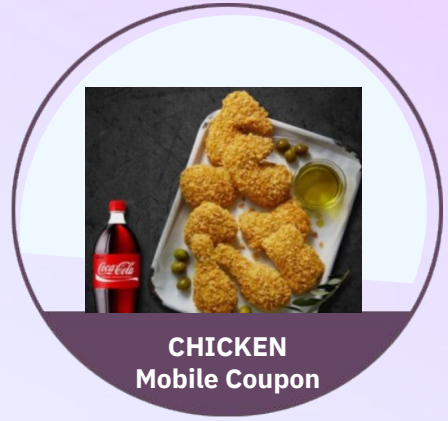
QUIZ 🤔

Q. 장애 상황을 신속하게 전파하는데 도움이 되는 Datadog 제품 2가지를 선택해 주세요

1. CI Visibility
2. Incident Management
3. Session Replay
4. On-Call

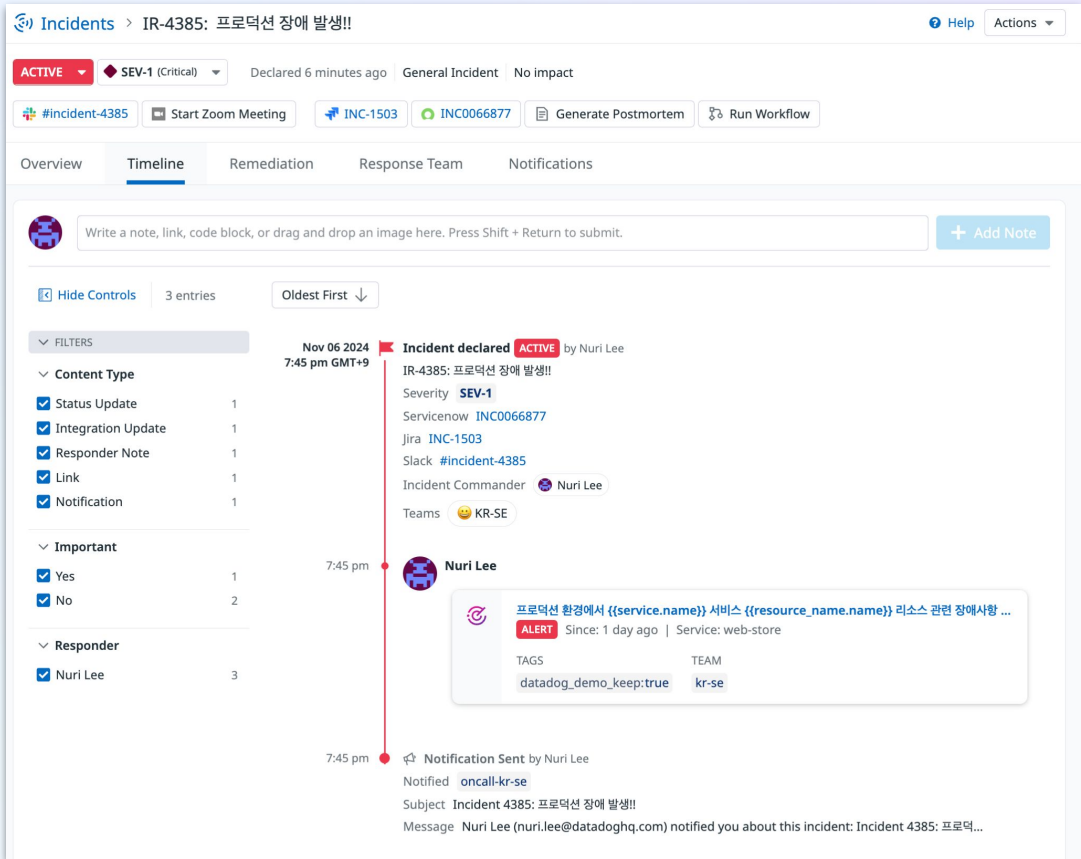
DATADOG

QUIZ 🤔



Q. 장애 상황을 신속하게 전파하는데 도움이 되는 Datadog 제품 이름 번호를 2가지 선택해 주세요

1. CI Visibility
2. **Incident Management**
3. Session Replay
4. **On-Call**



QUIZ 🤔

Q. Datadog의 대시보드에서 클라우드 자원을 제어할 수 있는 솔루션은?

ex) 낭비되고 있는 EC2 재시작/삭제 등

1. Network Performance Monitoring (NPM)
2. Synthetics Monitoring & Testing
3. Code Analysis
4. App Builder

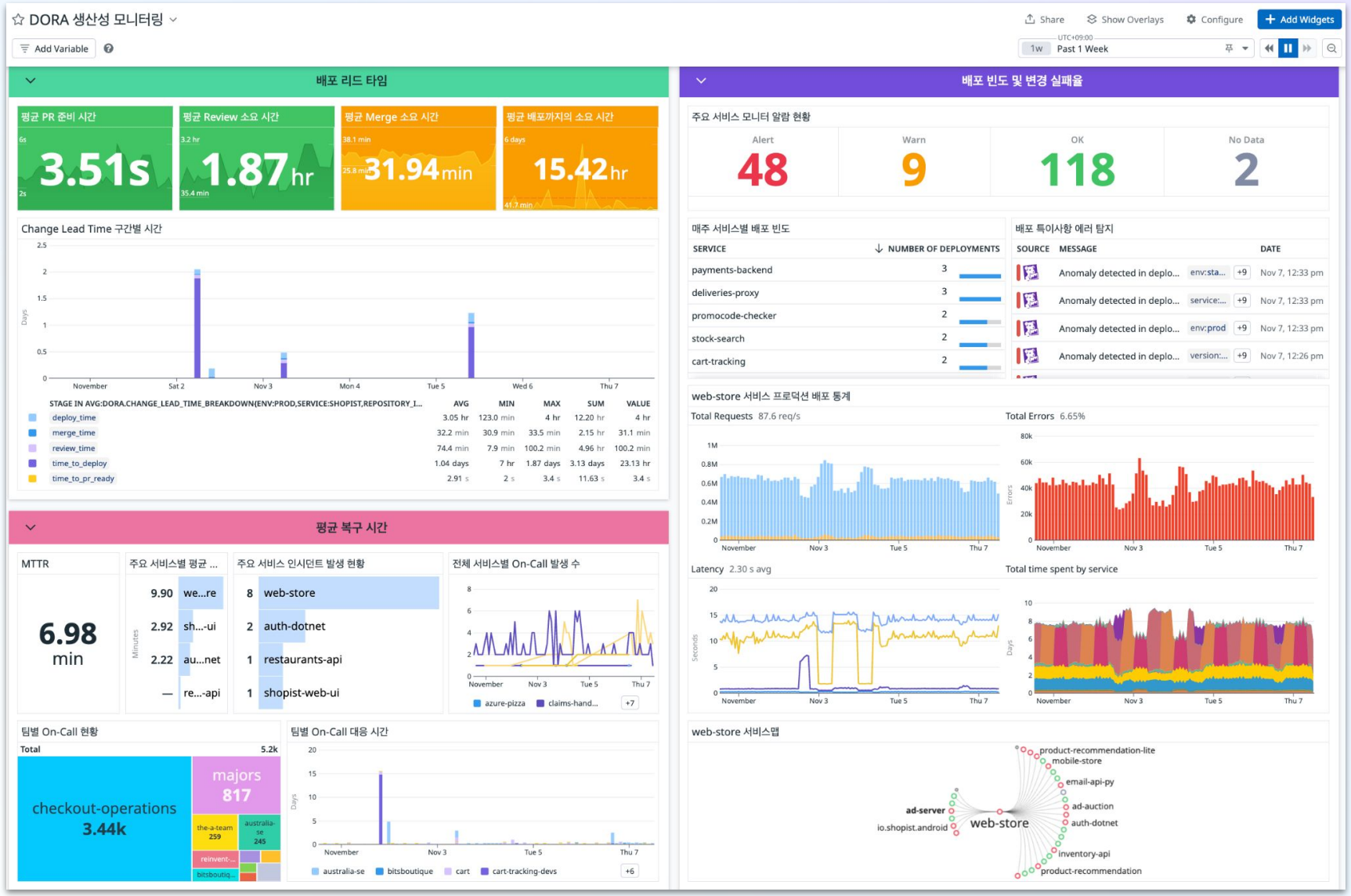
QUIZ 🤔

Q. Datadog의 대시보드에서 클라우드 자원을 제어할 수 있는 솔루션은?

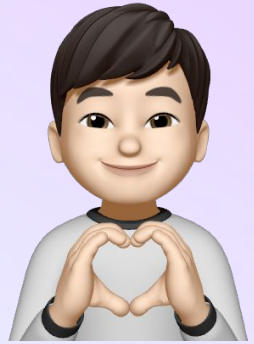
ex) 낭비되고 있는 EC2 재시작/삭제 등

1. Network Performance Monitoring (NPM)
2. Synthetics Monitoring & Testing
3. Code Analysis
4. **App Builder**

DORA 메트릭 End to End 대시보드 



DevOps를 잘하기 위한 1장 요약 ! 📝



✅ DORA 메트릭은 DevOps의 성숙도를 높이기 위해 꼭 필요한 기준

✅ Datadog APM, Github Integration, API를 통해 간단하게 메트릭을 수집

✅ DORA 메트릭 개선을 위한 솔루션들 💡

변경 리드 타임 배포 빈도	CI Visibility	CD Visibility	
변경 실패	Code Analysis Quality Gate	Test Optimization	APM (Deployment Tracking)
장애 복구 시간	Watchdog AI	Workflow Automation	Incident Management On-Call

Q&A

Q&A

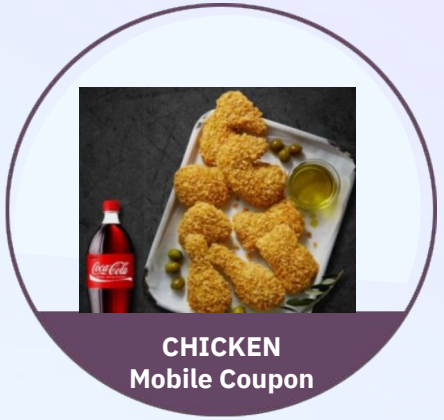
-

Thank You
감사합니다 ⭐

QUIZ 🤔

Q. 다음 중 Workflow Automation의 트리거 시작이 되는 컴포넌트는?

1. 알람 모니터 알람
2. Incident 장애 발생
3. 스케쥴링
4. 보안 위험 시그널 발생
5. 대시보드
6. 모두 다



DATADOG

QUIZ 🤔

Q. 다음 중 Workflow Automation의 트리거 시작이 되는 컴포넌트는?

1. 알람 모니터 알람
2. Incident 장애 발생
3. 스케쥴링
4. 보안 위험 시그널 발생
5. 대시보드
6. **모두 다**