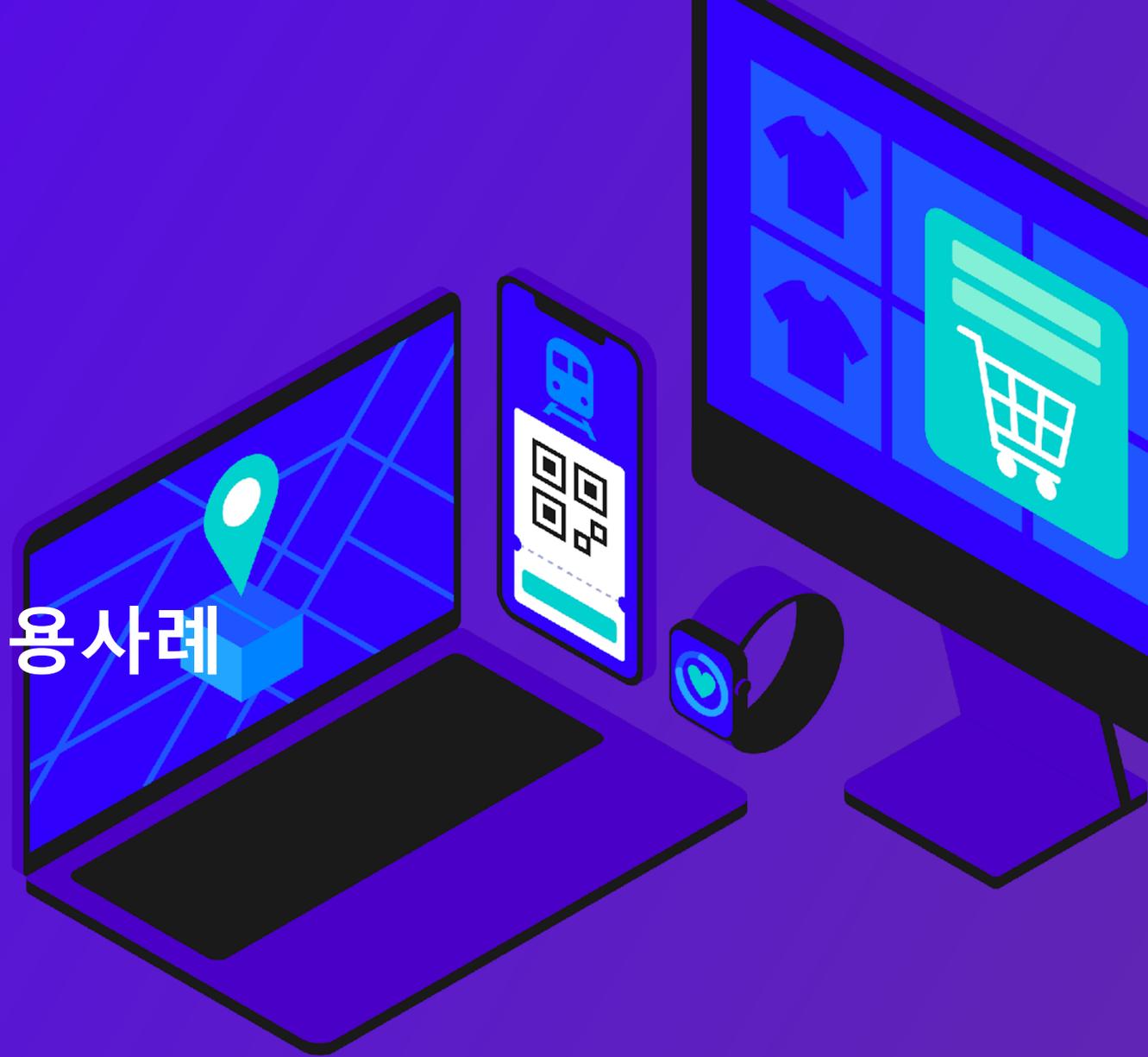


아모레퍼시픽: 뷰티 이커머스 테크 기업의 고도화/안정화를 위한 Front-End Observability 적용사례

정태인, 김성곤
디지털서비스개발팀
아모레퍼시픽



Speaker



정태인

아모레퍼시픽
디지털서비스개발팀 FE 개발



김성곤

아모레퍼시픽
디지털서비스개발팀 FE 개발

“

웹페이지 로딩이 100 ms
지연될 때마다 전환율이 7% 감소하며,
온라인 쇼핑몰에서 로딩시간이
3초 이상 걸리면
고객의 53%가 떠날 가능성이 있습니다.

”

서비스 장애와 충성고객 간 상관관계

Akamai Technologies' research

장애-서비스-고객의 위험한 삼각관계

장애를 예측하고 컨트롤 하는 것, **Observability**

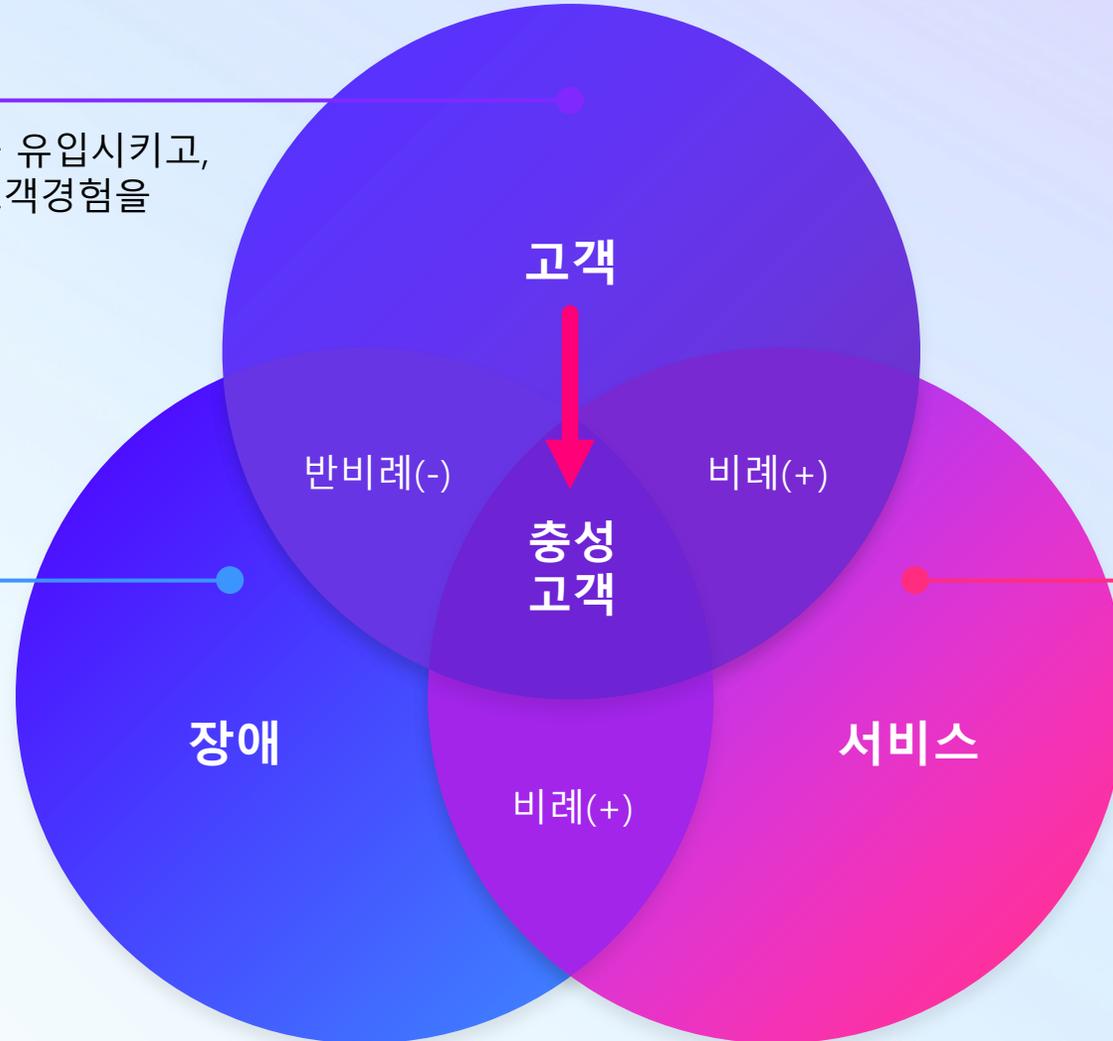
충성고객으로의 전환 조건

다양한 서비스를 통해 신규고객을 유입시키고, 장애를 최소화하여 만족스러운 고객경험을 제공해야 충성고객으로 전환됨

장애 관리의 중요성

서비스/고객/장애 중 우리가 컨트롤 할 수 있는 건 오직 장애 요소 뿐.

장애를 예측하고 컨트롤 해야함



서비스의 역설

서비스가 다양해져야 신규 고객의 유입이 비례하나 예측하지 못한 장애도 동시에 비례함

Agenda

01 아모레퍼시픽/아모레몰 소개

02 아모레몰의 성장 전략 및 데이터독 도입 배경

03 로그툴 도입을 위한 고려사항 및 데이터독 기능분석

04 데이터독 도입 내용 및 상세 설정

05 데이터독 도입 후 변화

01. 소개

아모레퍼시픽/아모레몰 소개

아모레퍼시픽

화장품 제조

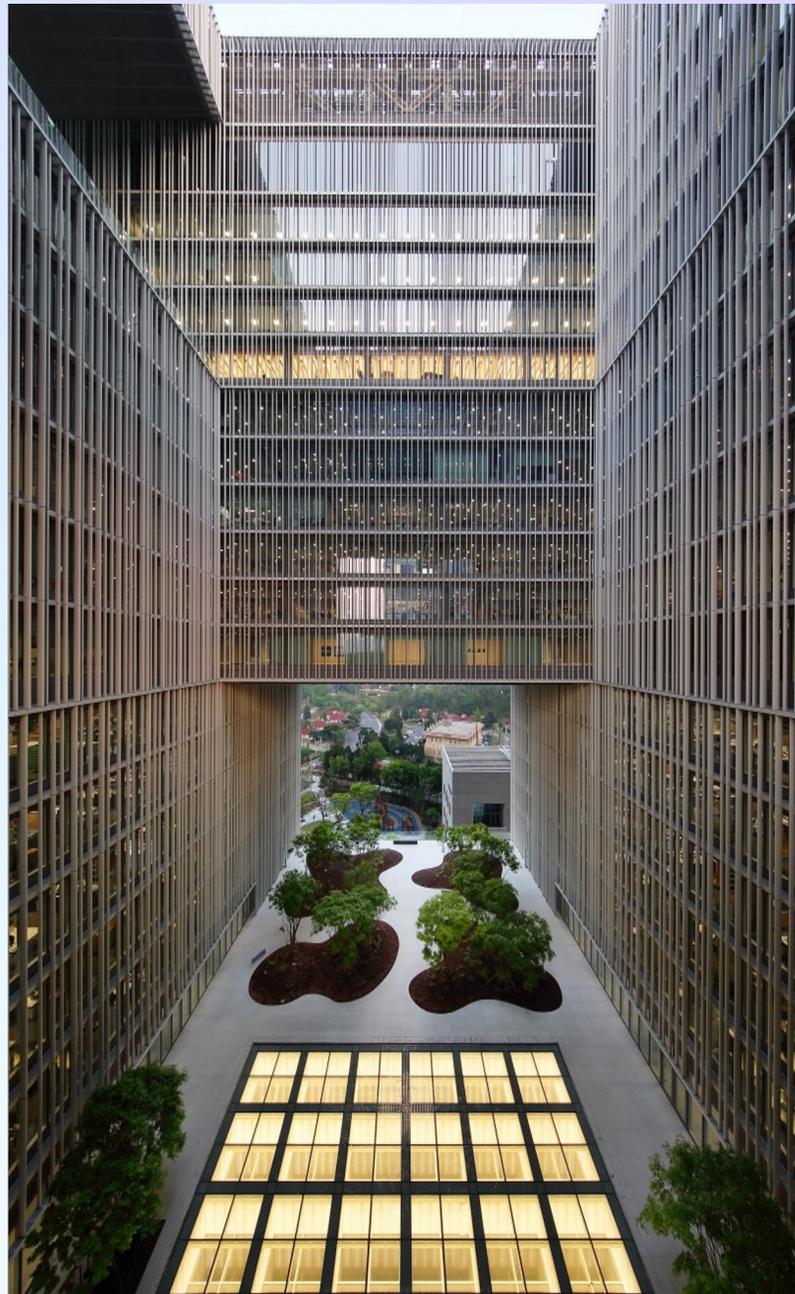
설화수, 이니스프리, 헤라, 한울,
프리메라, 마몽드, 비레디 등

+ 이커머스 유통

아모레몰, 이니스프리몰,
오설록몰, 글로벌 이커머스 등

+ AI 등 테크

스킨노트, 뷰티피드, 아모레스토어
등 고객 경험을 통해
얻은 데이터를 토대로
최적의 추천상품과 서비스를 제공

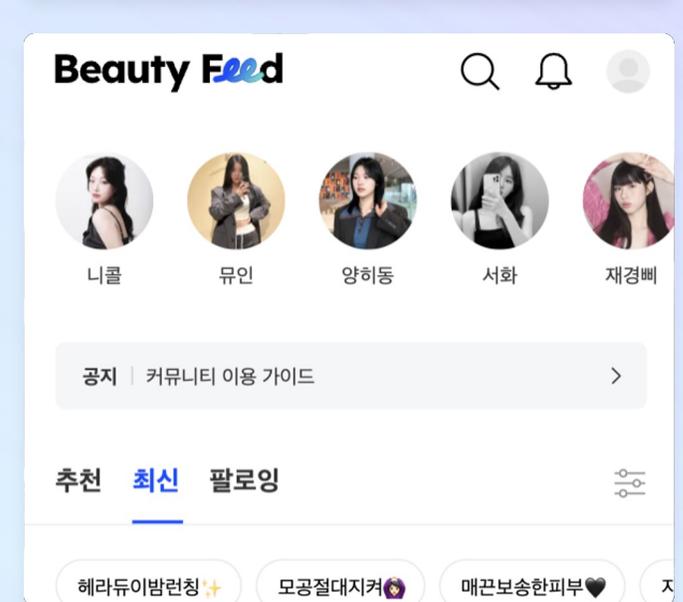
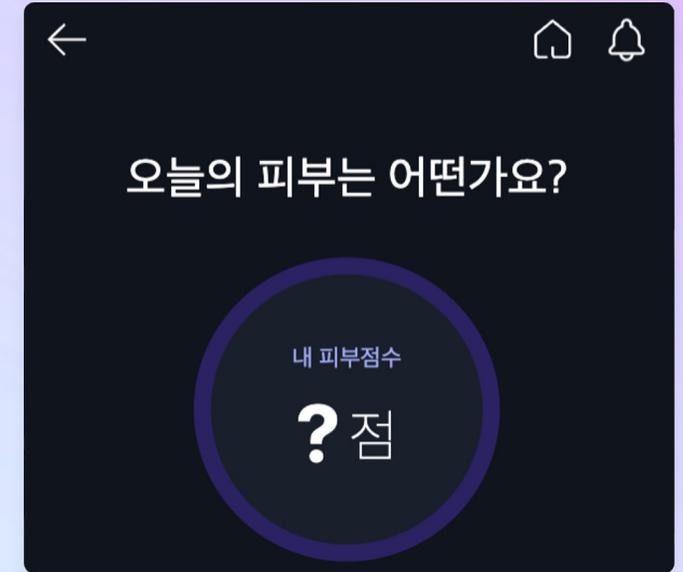
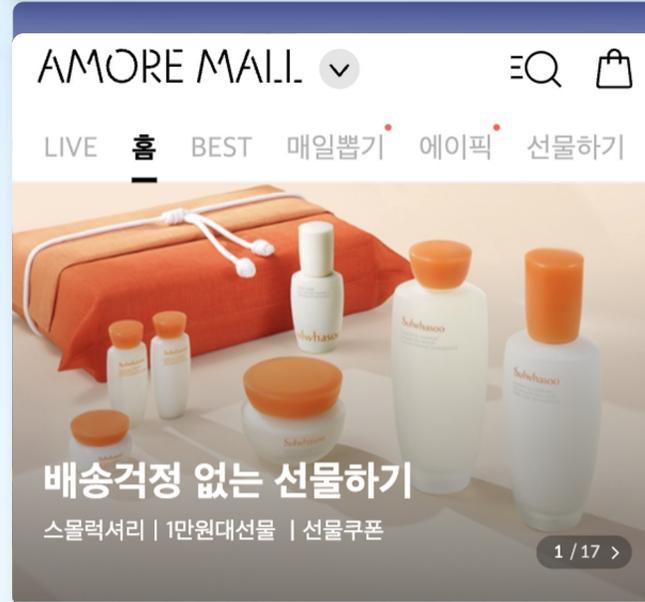


아모레몰

www.amoremall.com

아모레몰은
아모레퍼시픽의
대표 이커머스 몰로써,
스킨노트/뷰티피드/O2O 등
뷰티 이커머스 몰을 대표하는
다양한 테크 서비스의
집약체입니다.

다양한 고객 서비스를 통해
고객과 소통하며
고객이 원하는 화장품을
최고의 만족감으로 체험할 수 있는
온라인-오프라인을 넘나드는
쇼핑몰



02. 도입 배경

아모레몰의 성장 전략과 데이터독을 도입한 배경

아모레몰 성장 전략

아모레몰 1.0

기존 느린 성능의 주요 서비스를
내재화를 통해 개선

아모레몰 2.0

“성능 개선”을 목표로 리팩토링을
통해 5배의 서비스 성능 개선

아모레몰 3.0

고도화된 서비스에서, 빠른 장애
인지 및 고품질의 고객 피드백을
제공하는 안정화된 시스템으로의
발전



03. 고려사항 및 기능분석

로그툴 도입을 위한 고려사항 및 데이터독 기능 분석

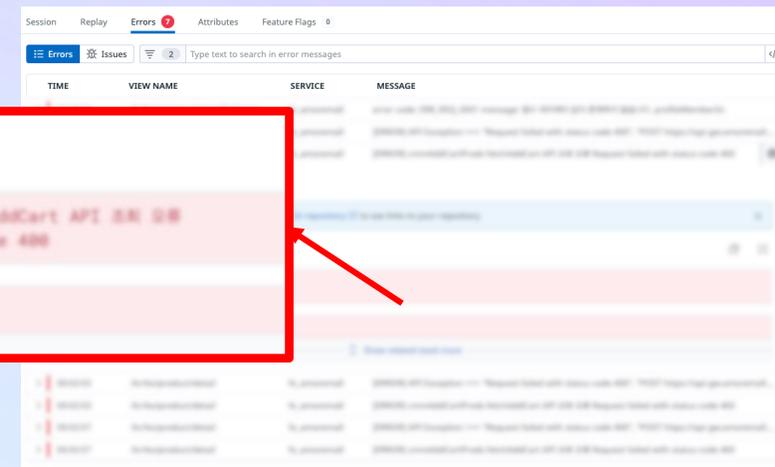
03. 고려사항 및 데이터독 기능분석

로그를 사용을 위한 고려사항 (1)

(1) 트러블슈팅 집중 여부 및 쉬운 편의성 *

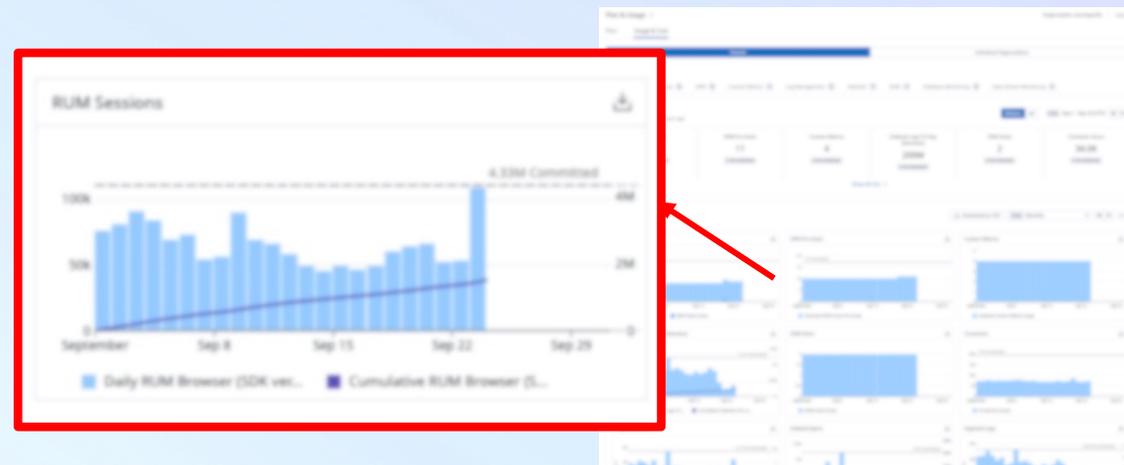
- 트러블 슈팅 상세 지원(RUM*, Log, Alarm 등)
- 소스맵을 통한 장애 코드 원인 분석 지원
- 직관적인 UI/UX, 기술지원/상세문서 제공

* 로그를 고려사항 중 제일 중요(장애 대응 및 관리)



(2) 운영을 위한 비용 효율성

- 도입예상비용과 ROI(Return On Investment) 계산
- 서비스별 비용조절 용이성
- 이벤트(가정의 달 행사 등)에 유연한 비용관리 제공



* RUM : Real User Monitoring 의 약어(이하 RUM)

03. 고려사항 및 데이터독 기능분석

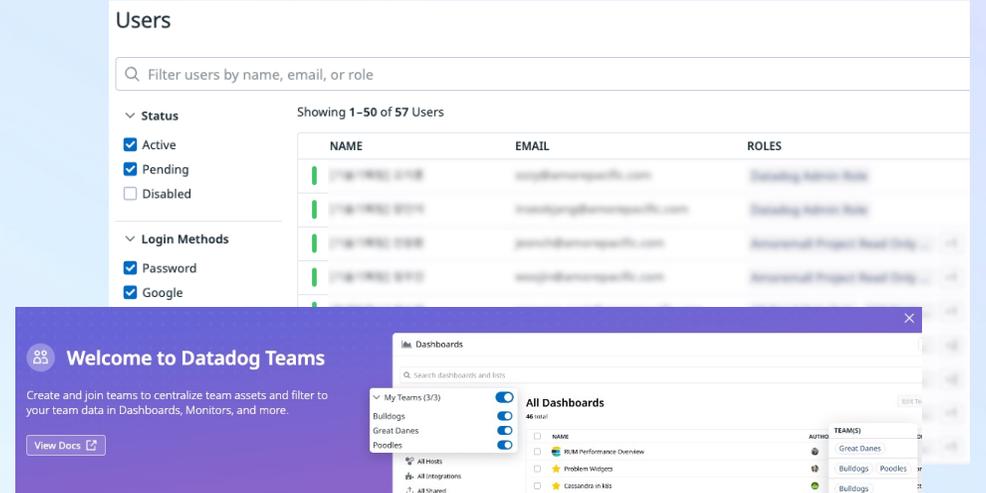
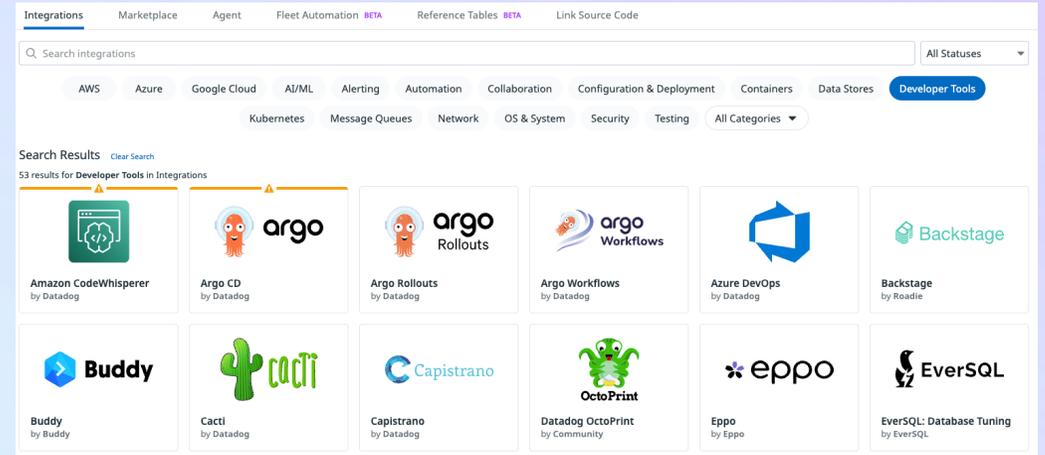
로그툴 사용을 위한 고려사항 (2)

(3) 통합성 및 확장성

- 서비스 성장에 따른 시스템 확장 가능성
- 다른 서비스 플랫폼과의 통합 가능성

(4) 사용자 및 역할 관리

- 다양한 사용자 및 역할별 권한 설정 가능 여부
- 권한 관리를 통해 데이터에 대한 접근 통제 여부



03. 고려사항 및 데이터독 기능분석

로그를 사용을 위한 고려사항 (3)

(5) 보안 및 암호화

- 고객의 민감정보를 고려한 민감정보 암호화 지원
- 소스코드를 소스맵 방식 등을 통한 유출방지 지원
- 사용자 권한 별 접근 제한 및 분류

(6) 데이터 분석 및 시각화

- 적재된 로그 및 고객행동 데이터를 기반으로 유의미한 서비스 분석
- A/B TEST 에 활용하여 고객경험 분석 지원
- 로그 데이터를 대시보드나 리포트 형태로 시각화

(7) 실시간 모니터링과 알림 *

- 실시간으로 로그를 모니터링
- 이상 징후, 장애발생 알람을 전송하는 기능
- 담당자가 고객보다 빠르게 인지하고 상세히 파악

* 로그를 고려사항 중 중요(장애를 빠르게 대응하고, 예측하기 위함)



03. 고려사항 및 데이터독 기능분석

데이터독 지원 기능분석



TroubleShooting

- 세션 기반 기능 지원
- 뷰 또는 에러 기반 기능 지원
- SourceMap Upload 기능 지원
- User Action Tracking 기능 지원
- 다양한 필터링으로 조회기능 지원
- 오류 로그에 대한 세분화된 표시 지원



Alarm

- WebHook 기능 지원
- WebHook 알람 레벨 세분화 기능
- WebHook 이 Jira/Slack 등과 연계
- WebHook 알람별 모니터링 세분화
- 필터링을 통한 알람 기능 분류 지원



Code Quality & Issue Check

- 코드 품질 필터링 세분화 지원
- 코드 품질 배포 버전별 관리 기능
- 고객행동 기반 액션 오류 파악 기능



Business Intelligence

- 대시보드의 세분화 지원
- 대시보드의 Ctrl C+V 등 사용 편의
- 대시보드의 고객친화적 디자인
- RUM/Log 기반 필터링을 B.I 지표에 적용

04. 데이터독 상세 설정

데이터독 Front-End 도입 내용 및 상세 설정
(RUM, Logs, Monitor, Dashboard)

RUM / Logs 초기화 모듈

Typescript & ES6

아모레몰 Front-End 개발 표준 준수

Imports

Datadog Browser SDK

Interfaces

데이터독 실행을 위한 Props 정의

Functions

RUM / Logs 초기화 실행 (datadog)

공통 설정 및 샘플링 레이트 조회

```

datadog/index.ts
├── Imports:
│   ├── @datadog/browser-rum           | Datadog Browser SDK
│   └── @datadog/browser-logs         | Datadog Browser SDK
├── Interfaces:
│   └── DatadogConfig                 | Datadog SDK 및 SDK 옵션
├── Functions:
│   ├── initializeDatadog             | Datadog SDK
│   ├── initializeDatadogLogs         | Datadog Logs SDK 및 SDK
│   ├── initializeDatadogRum         | Datadog RUM SDK 및 SDK
│   ├── getCommonConfig               | SDK Datadog SDK
│   └── getConditionalSampleRate      | RUM & Logs sample rate SDK

```

RUM / Logs 초기화 모듈

initializeDatadog 함수 호출

RUM 및 logs 초기화 실행
_app.tsx 최상단 호출(Next.js 기준)

datadogRum.setUser 설정

식별 및 구별 가능한 세션 정보

```
// initialize datadog
useEffect( effect: () :void => {
  const pathMap: NextPathNameMap = pathmap;
  const pathInfo = pathMap[router.pathname] ?? {};

  initializeDatadog({
    applicationId: publicRuntimeConfig.applicationId,
    clientToken: publicRuntimeConfig.clientToken,
    service: publicRuntimeConfig.service,
    version: publicRuntimeConfig.version,
    isActiveLog: publicRuntimeConfig.isActiveLog,
    isActiveRum: publicRuntimeConfig.isActiveRum,
    env: publicRuntimeConfig.env,
    excludedErrorLogIds: publicRuntimeConfig.excludedErrorLogIds,
    isAllowCollectingTelemetry: publicRuntimeConfig.isAllowCollectingTelemetry,
  });

  export const initializeDatadog = (config: DatadogConfig) :void => {
    if (typeof window !== 'undefined') {
      if (config.isActiveRum && config.isActiveLog) {
        return;
      }
      try {
        initializeDatadogLogs(config);
        initializeDatadogRum(config);
      } catch (e) {
        logger.error(e, {message: 'datadog initializeDatadog RUM OR LOG'});
      }
    }
  };

  datadogRum.setUser({
    // user: {id: 'user', name: 'user', email: 'user@naver.com', phone: '010-1234-5678'},
  });
}, []);
```

Datadog RUM 초기화 설정

allowedTracingUrls 설정

RUM을 통한 API trace 지원

sample rate 설정

불필요 세션에 대한 샘플레이트 조절

tracking 옵션 활성화

사용자 행동 분석 제공

beforeSend: tag name filtering

민감성 정보 수집 제외

```
const initializeDatadog = (config: DatadogConfig) => () => {
  if (config.isActive()) {
    datadogum.init({
      ...getCommonConfig(config),
      beforeSend: (g: RUMEvent) => {
        if (g.type === 'action') {
          const isIncludeAction =
            !P._.action?.target ||
            !isIncludeAction(g.action.target.name)
        }
      }
    })
  }
  return true;
};

export const config = {
  allowedTracingUrls: [https://v.vip-get.amoreall.com/],
  sessionSampleRate: getConditionalSampleRate(),
  trackUserInteractions: true,
  trackResources: true,
  trackLongTasks: true,
};

logger.debug('datadog rum initialized');
```

```
allowedTracingUrls: [https://v.vip-get.amoreall.com/],
sessionSampleRate: getConditionalSampleRate(),
trackUserInteractions: true,
trackResources: true,
trackLongTasks: true,
```

```
allowedTracingUrls: [https://v.vip-get.amoreall.com/],
sessionSampleRate: getConditionalSampleRate(),
trackUserInteractions: true,
trackResources: true,
trackLongTasks: true,
```

Datadog RUM 초기화 설정

allowedTracingUrls 설정

RUM을 통한 API trace 지원

sample rate 설정

불필요 세션에 대한 샘플레이트 조절

tracking 옵션 활성화

사용자 행동 분석 제공

beforeSend: tag name filtering

민감성 정보 수집 제외

```
const initializeDatadog = (config: DatadogConfig) => () => {
  if (config.isActiveRum) {
    datadogRum.init({
      ...getCommonConfig(config),
      beforeSend: (g: RumEvent) => {
        if (g.type === 'action') {
          const isIncludeActionName: boolean =
            !g._dd.action?.target?.selector?.includes('data-dd-action-name') ?? false;

          if (!isIncludeActionName && config?.isAllowCollectingTagNames && g.action?.target?.name) {
            g.action.target.name = g._dd.action?.target?.selector || g.action.target.name.replace(/.*/g, '*');
          }
        }
      },
    });
  }
  return true;
};
```

```
applicationId: config.applicationId,
allowedTracingUrls: [https://*.api-gel.amerapac.com/],
sessionSampleRate: getConditionalSampleRate(),
trackingInteractions: true
});
```

필터링 우선순위: dd-action-name > page > tag selector > default(*)

```
beforeSend: (g: RumEvent) => {
  if (g.type === 'action') {
    const isIncludeActionName: boolean =
      !g._dd.action?.target?.selector?.includes('data-dd-action-name') ?? false;

    if (!isIncludeActionName && config?.isAllowCollectingTagNames && g.action?.target?.name) {
      g.action.target.name = g._dd.action?.target?.selector || g.action.target.name.replace(/.*/g, '*');
    }
  }
  return true;
};
```

Datadog RUM 초기화 설정

allowedTracingUrls 설정

RUM을 통한 API trace 지원

sample rate 설정

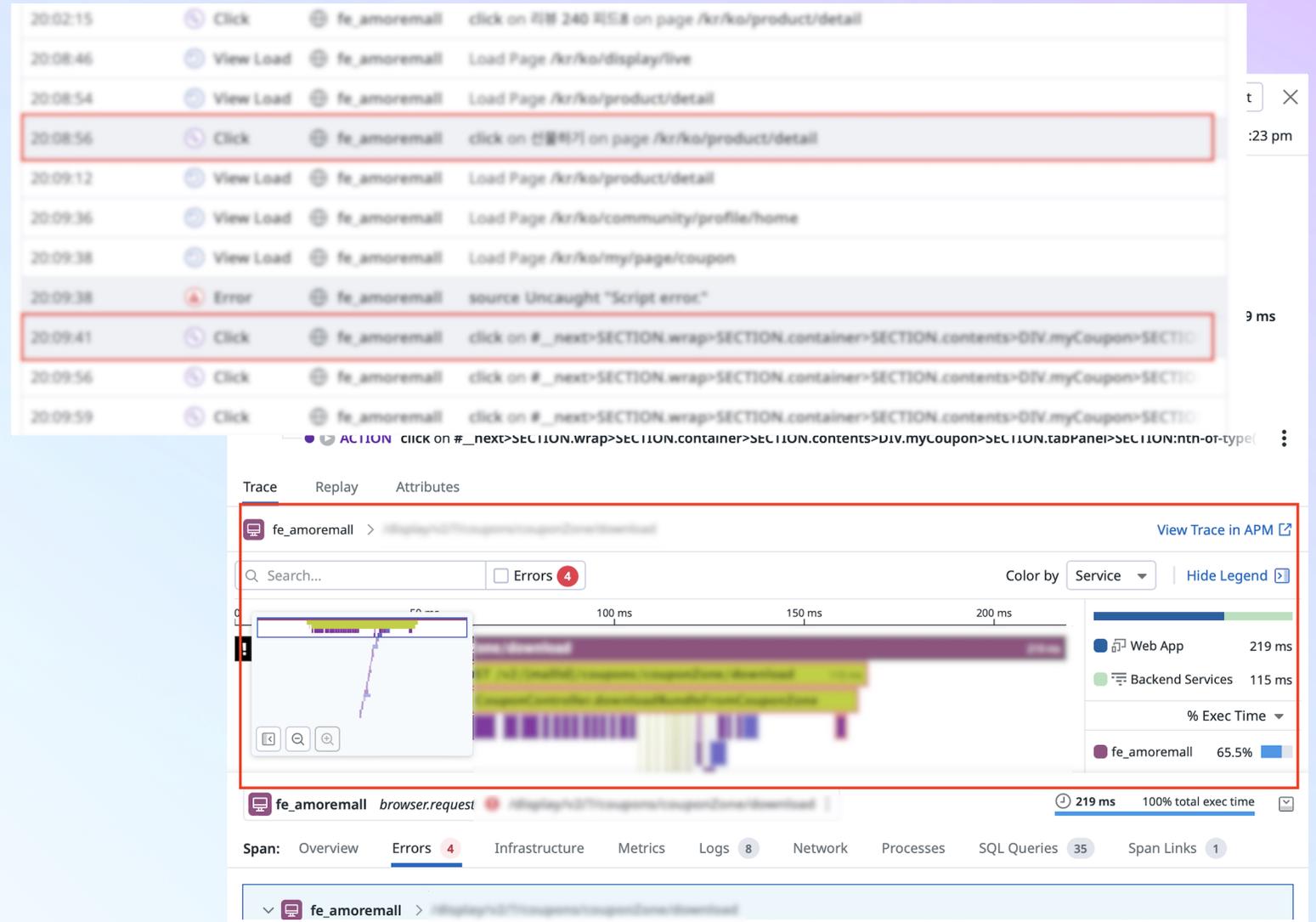
불필요 세션에 대한 샘플레이트 조절

tracking 옵션 활성화

사용자 행동 분석 제공

beforeSend: tag name filtering

민감성 정보 수집 제외



Datadog Logs 초기화 설정

application Error log

forwardErrorToLogs 활성화
console.error / logger.error

beforeSend: 로그 수집 제외

origin type 'network'
excludedErrorLogList

Amore Logger : monitor log

모니터링용 신규 타입 적용
origin type 'monitor'

```
const initializeDatadogLogs = (config: DatadogConfig) => () => {
  if (config.isActiveLog) {
    datadogLogs.init({
      ...getCommonConfig(config),
      forwardErrorsToLogs: true,
      beforeSend: (g: LogEvent) => {
        if (g.status === 'error') {
          if (g.origin === 'network' && config.excludedErrorLogList?.find((log: string) => g.message.includes(log)) <
            return false;
          }
        }
        return true;
      },
    });
    logger.debug('Datadog log initialized');
  }
};
```

```
export const excludedDataDogErrLogs :string[] = [
  // 마케팅 스크립트 관련 오류
  'Uncaught "Script error."',
  'The database connection is closing.',
  'One of the specified object stores was not found.',
  '%c [AIRBRIDGE] TypeError:',
  'Rollbar:',
  'criteo.q is not defined',
  // 라이브 외부 스크립트 관련 오류
  'AxiosError: Network Error',
  'Loading chunk',
  'find variable: webChatClose',
  'find variable: isWifi',
```

Datadog Logs 초기화 설정

application Error log

forwardErrorToLogs 활성화
console.error / logger.error

beforeSend: 로그 수집 제외

origin type 'network'
excludedErrorLogList

Amore Logger : monitor log

모니터링용 신규 타입 적용
origin type 'monitor'

```
monitor: (( level, message, args ): { level: LogLevel; message: string; args?: LoggableArg[] }) => {
  if (!isServer) {
    console.error('log.monitor() function is allowed to be declared by the client only.');
```

```
    } else {
      const metadata: Metadata | undefined = args?.reduce((agg: Metadata, current: LoggableArg, idx: number) => {
        if (current != null && current != undefined) {
          const key: string = `attribute${idx}`;
          agg[key] = current;
        }
      }, {});
      return agg;
    }
  }, {});
```

The screenshot shows a Datadog Logs interface. A log entry is expanded, showing details such as application ID, date, request ID, log level, and message details. The 'origin' field is highlighted in a red box, and a red arrow points from the code snippet above to this field, indicating that the log entry was generated by the 'monitor' origin.

Datadog App Monitor

Monitor 생성

- 페이지 호출량 증가
- 동시 접속자 증가
- 어뷰징 의심 사용자
- 페이지 및 API 응답 이상

Alarm 발송 (Webhook)

- 업무 메신저 연동을 통해 알람 발송
- 3단계 구분 알람 발송(경고, 주의, 해제)



Datadog App Monitor

Monitor 생성

- 페이지 호출량 증가
- 동시 접속자 증가
- 어뷰징 의심 사용자
- 페이지 및 API 응답 이상

Alarm 발송 (Webhook)

- 업무 메신저 연동을 통해 알람 발송
- 3단계 구분 알람 발송(경고, 주의, 해제)

Configure notifications & automations

[RUM][AMORE MALL] API GW 응답 오류 발생

@@ {{#is_alert}}[경고][AMORE MALL] API GW 호출 오류 1분간 50건 이상 발생
 - 호출 경로: {{{@resource.url_path_group}.name}}
 - 발생 건수: {{{eval "int(value)"}}}
 {{{/is_alert}}}{ {{#is_warning}}}[주의][AMORE MALL] API GW 호출 오류 1분간 30건 이상 발생
 - 호출 경로: {{{@resource.url_path_group}.name}}
 - 발생 건수: {{{eval "int(value)"}}}
 {{{/is_warning}}}

@webhook-feamoremall

Variables

VARIABLE	VALUE
\$RED_CONNECT_COLOR	#DC143C
\$GREEN_CONNECT_COLOR	#3E8245
\$YELLOW_CONNECT_COLOR	#E1BC3A

Webhooks

feamoremall

URL: https://api.feamoremall.com/api/v1/alerts/webhook

PAYLOAD

```
{
  "body": "SEVENT_TITLE",
  "connectColor": "$RED_CONNECT_COLOR",
  "connectInfo": [
    {
      "description": "SEVENT_MSG"
    }
  ]
}
```

datadog monitor

[Triggered] [RUM][AMORE MALL] API GW 응답 오류 발생

[[[경고][AMORE MALL] API GW 호출 오류 1분간 50건 이상 발생
 - 호출 경로: {{{@resource.url_path_group}.name}}
 - 발생 건수: 102
 The monitor was last triggered at Wed Sep 25 2024 13:46:50 KST.]]]]

[Warn] [RUM][AMORE MALL] API GW 응답 오류 발생 (통합)

[[[주의][AMORE MALL] 5분간 API GW 호출 오류 200건 이상 발생(136)
 The monitor was last triggered at Wed Sep 25 2024 13:47:11 KST.]]]]

[Recovered] [RUM][AMORE MALL] API GW 응답 오류 발생

Datadog App Dashboard

FE 통합 모니터링 대시보드

실시간 페이지 요청량

- 통합페이지
- 5대 주요 페이지

실시간 에러 발생량

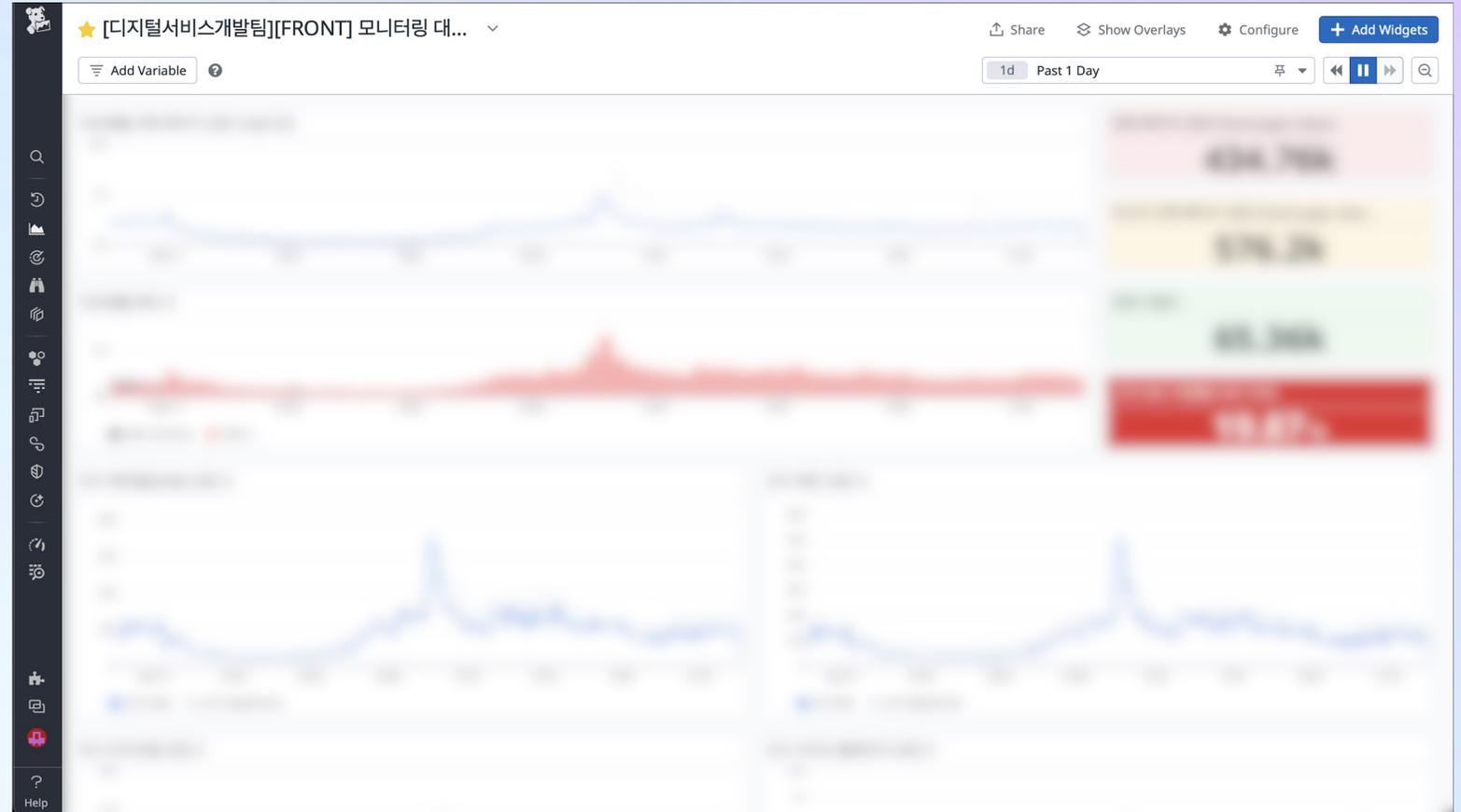
동시 사용자 및 세션 이탈률

에러 모니터링 대시보드

실시간 에러 모니터링

- 전체 에러 모니터링
- 페이지/액션 에러 모니터링
- 배포 버전별 에러 모니터링

주요 에러 모니터링



Datadog App Dashboard

FE 통합 모니터링 대시보드

실시간 페이지 요청량

- 통합페이지
- 5대 주요 페이지

실시간 에러 발생량

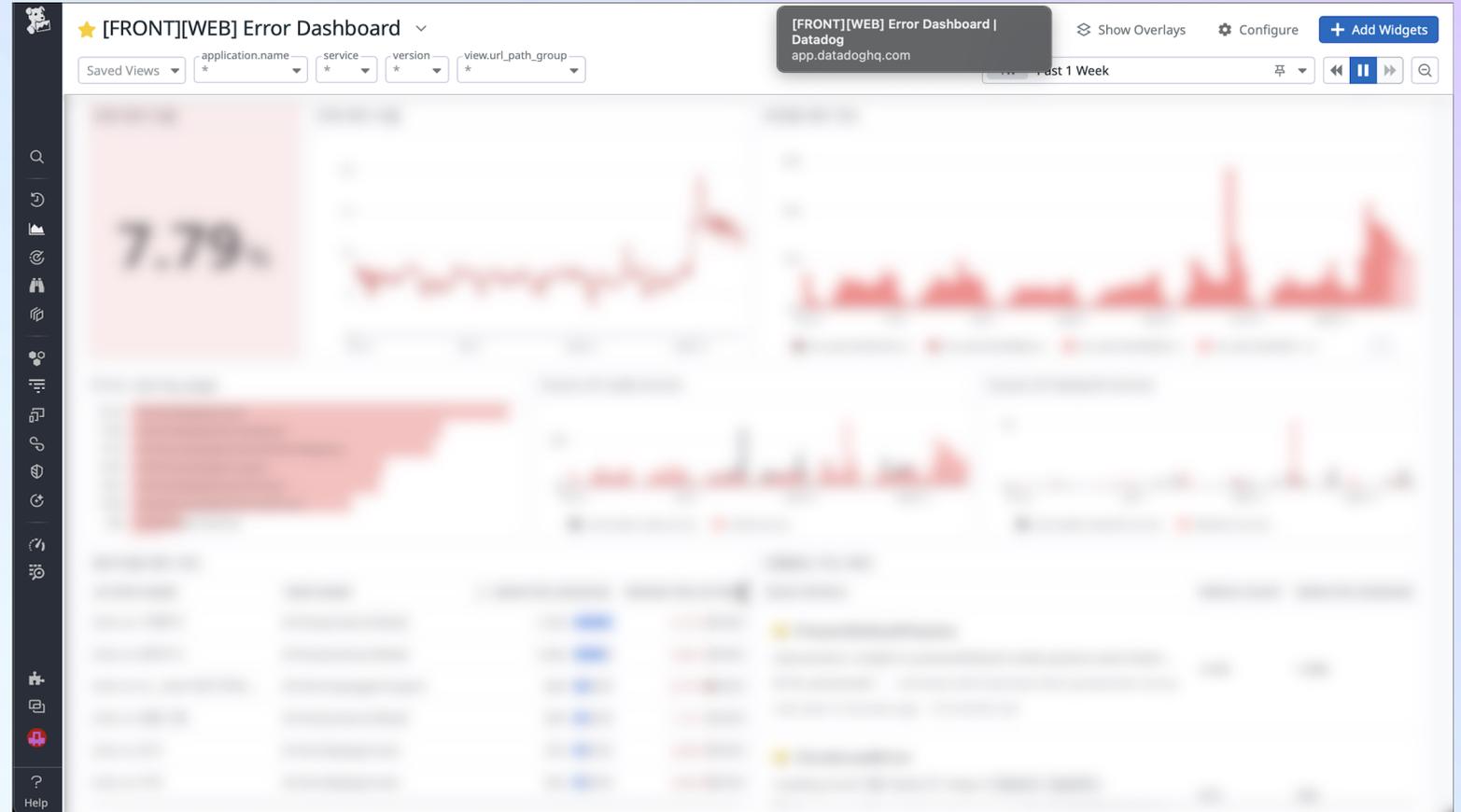
동시 사용자 및 세션 이탈률

에러 모니터링 대시보드

실시간 에러 모니터링

- 전체 에러 모니터링
- 페이지/액션 에러 모니터링
- 배포 버전별 에러 모니터링

주요 에러 모니터링



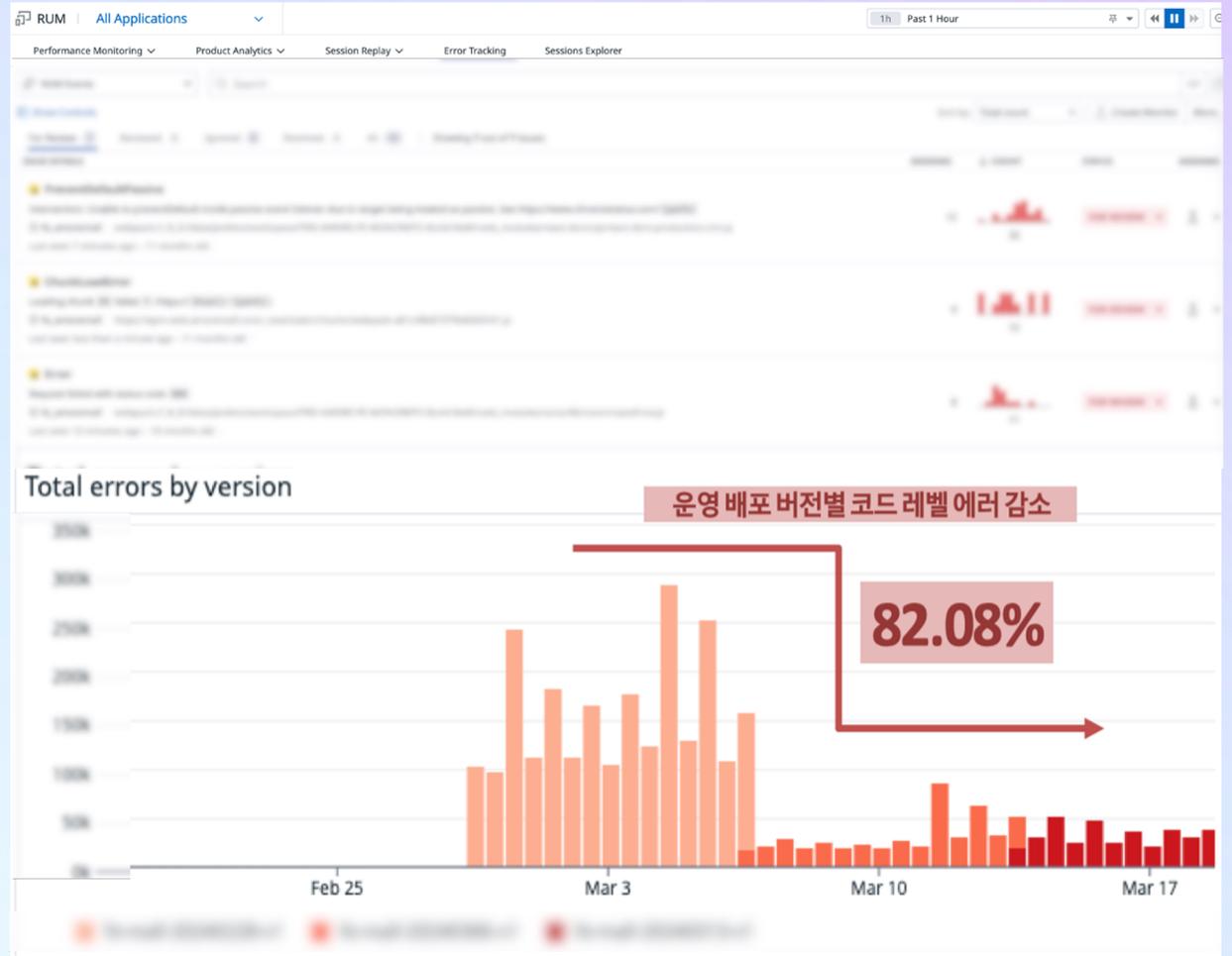
05. 데이터독 도입 후 변화

주요 개선점 및 활용사례

05. 데이터독 도입 후 변화

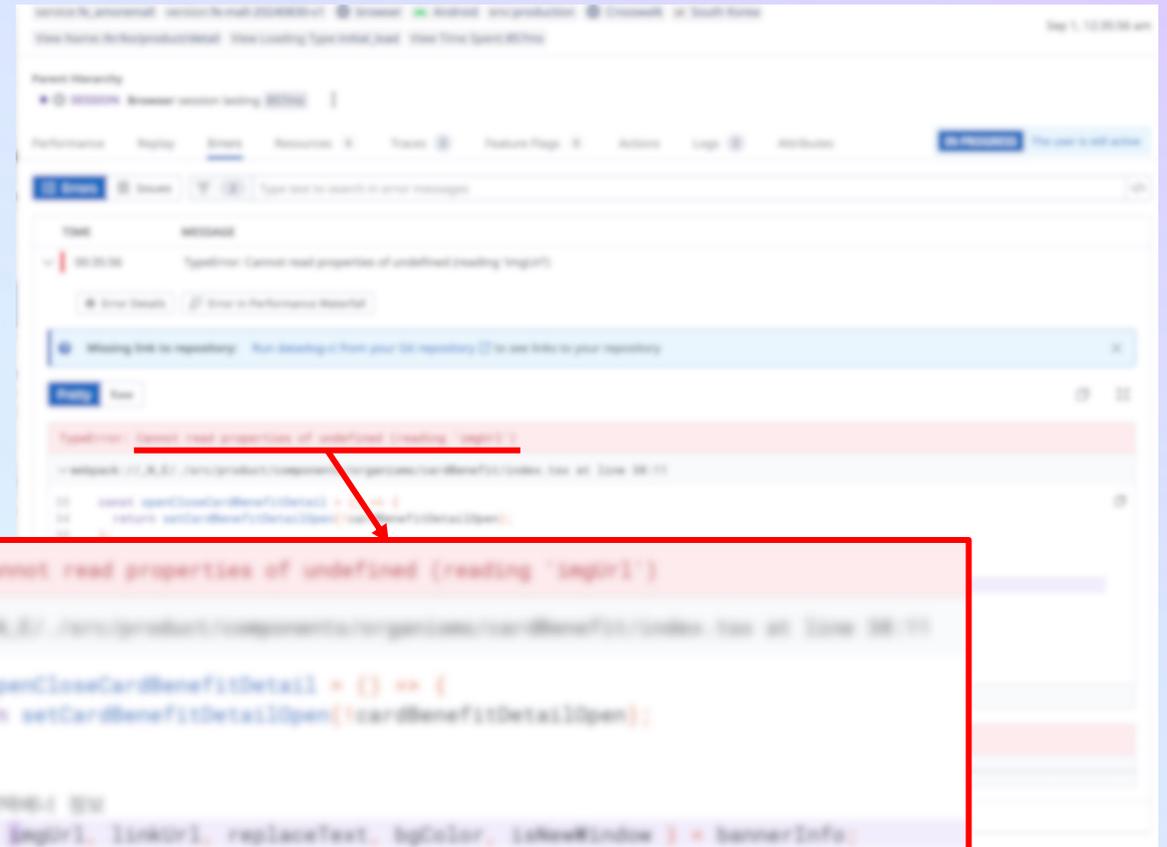
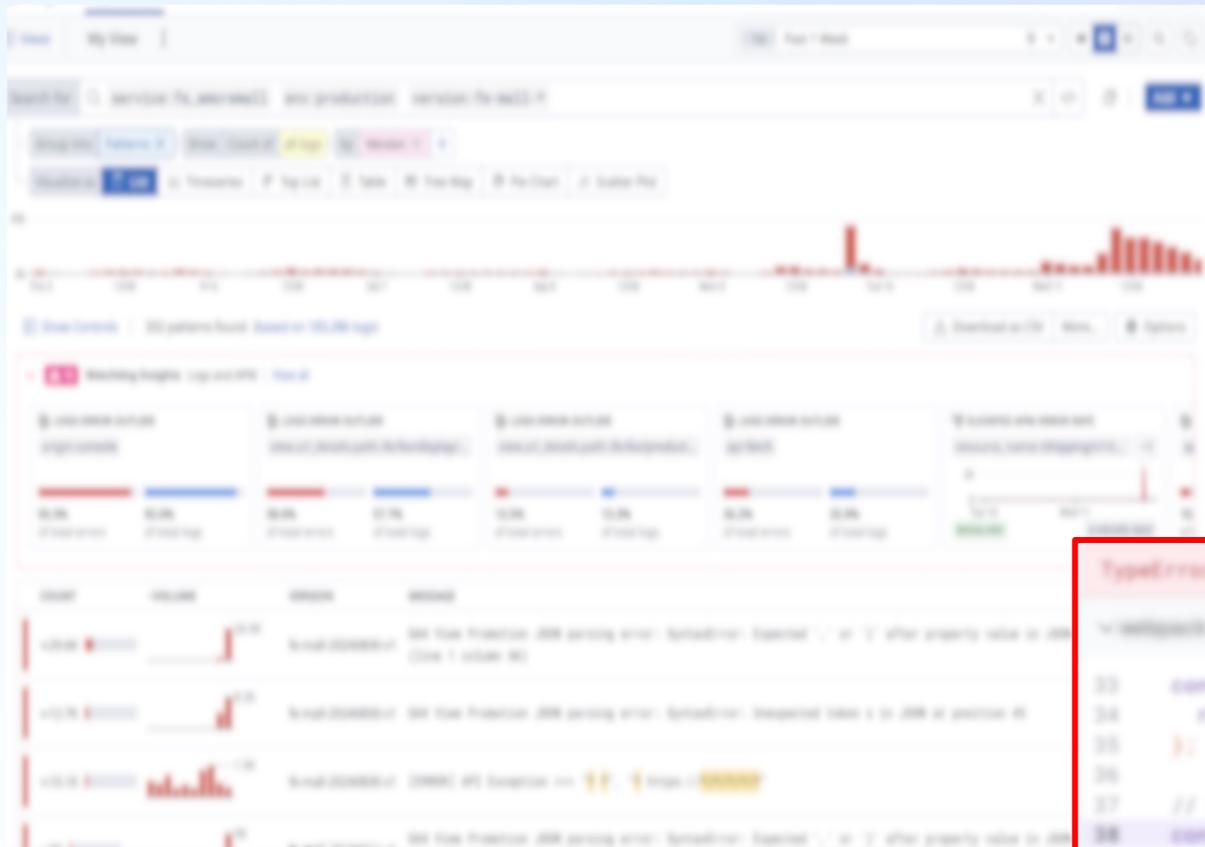
+82%

RUM, Log Management 기반으로
배포 버전별 관리를 통해
에러 코드 개선 및 코드 품질 관리



05. 데이터독 도입 후 변화

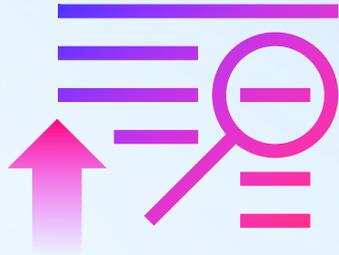
실시간 에러 모니터링을 통한 오류 원인 확인 및 장애 대응



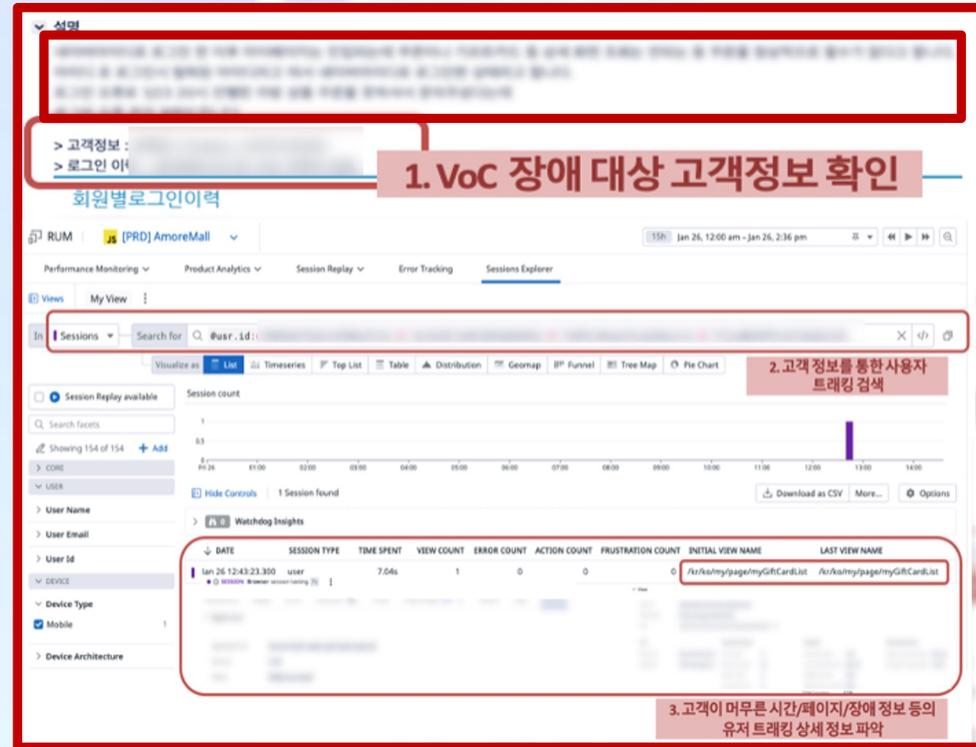
```
TypeError: Cannot read properties of undefined (reading 'imgurl')

webpack -- 3.21 - /src/product/components/organisms/cardBenefit/index.js at Line 38:11
33   const openCloseCardBenefitDetail = () => {
34     return setCardBenefitDetailOpen(!cardBenefitDetailOpen);
35   };
36
37   // FIXME: fix
38   const [imgurl, linkurl, replaceText, bgColor, showWindow] = bannerInfo;
39   // FIXME: fix
40   const showCardBenefitBanner = useShowArea(imgurl);
41   // FIXME: All this is done
42   const [showAltText, setShowAltText] = useState-boolean(false);
43
```

05. 데이터독 도입 후 변화



RUM 을 통해
고객의 행동 정보를 바탕으로
네트워크/서비스장애 및 에러 범위 등의
원인을 다각적으로 분석하여
빠른 VoC 응대와 대응 품질을 높임



1. VoC 장애 대상 고객정보 확인

2. 고객정보를 통한 사용자 트래킹 검색

DATE	SESSION TYPE	TIME SPENT	VIEW COUNT	ERROR COUNT	ACTION COUNT	FRUSTRATION COUNT	INITIAL VIEW NAME	LAST VIEW NAME
Jan 26 12:43:23.300	user	7.04s	1	0	0	0	/r/ko/my/page/myGiftCardList	/r/ko/my/page/myGiftCardList

3. 고객이 머무른 시간/페이지/장애 정보 등의 유저 트래킹 상세 정보 파악

05. 데이터독 도입 후 변화

사용자 행동 정보를 기반으로 정확한 오류 원인 분석

The image displays a screenshot of the Datadog monitoring interface. On the left, a blurred view of the error log shows a sequence of events. A red box highlights a specific error: 'console [ERROR] API Exception === "Request failed with status code 400", "GET https://api-...'. On the right, a more detailed view of the 'SPA Route Change' event is shown. A red box highlights a 'frustration detected' event with the following details: 'service: fe_amoremail', 'version: fe-mail-20240830-v1', 'browser', 'Android', 'env: production', and 'Chrome...'. Below this, the 'Parent Hierarchy' shows 'SESSION Browser session lasting 12m'. At the bottom, the 'Errors' tab is active, showing 82 errors, and a search bar is visible with the text 'Type text to search in error messages'. The interface also shows 'Performance', 'Replay', 'Resources (99+)', 'Traces (15)', 'Feature Flags (0)', 'Actions (99+)', and 'Logs (8)'.

05. 데이터 독 도입 후 변화

기타 사용자 수집 데이터를 통한 원인 분석



A screenshot of a Datadog interface showing a table of view data. The table has four columns: 'VIEW NAME', 'VIEW REFERRER', 'COUNTRY', and 'BROWSER NAME'. A red box highlights a portion of the table, and a red arrow points from the log viewer to this table.

VIEW NAME	VIEW REFERRER	COUNTRY	BROWSER NAME
https://www.instagram.com		Japan	Mobile Safari
https://www.instagram.com	https://m.search.naver.com	Japan	Samsung Internet
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Canada	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari UI/WKWebView
https://www.instagram.com	https://www.instagram.com	Japan	Instagram
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Japan	Mobile Safari
https://www.instagram.com	https://www.instagram.com	Canada	Samsung Internet
https://www.instagram.com	https://www.instagram.com	Canada	Samsung Internet

Thank you



DATADOG