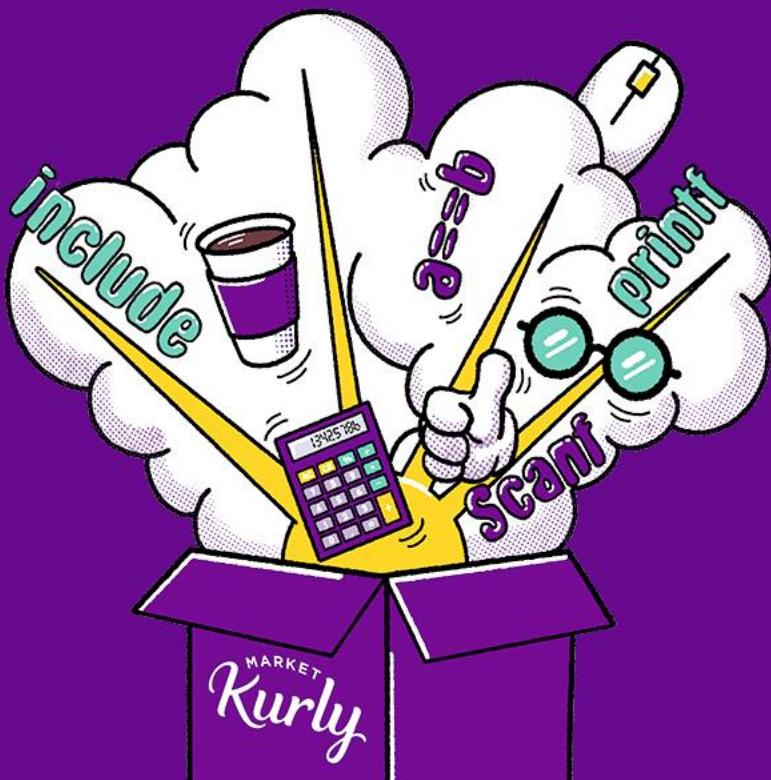


# MARKET KURLY DEVELOPER MEET UP

FUN 하지만 뻔하지 않은, 컬리의 개발자 성장기



# 컬리 개발자 성장기

컬리 도메인별 현업 개발자의 성장 스토리를 소개합니다.





컬리 개발자성장기

## 첫번째 이야기의 주인공

저는

“컬리에서 상품을 구매하는 과정이 즐거울 수 있게 만드는 개발자”

김민중 입니다.



> 김민중님 | 커머스플랫폼 웹 개발

[컬리에서 서비스와 함께 자라기]

“컬리 웹프론트 개발 조직엔 흥미로운 문제 정의들이 많습니다.  
e-commerce 서비스에서 레거시를 걷어내고,  
새로운 시스템을 만들어가는 성장 이야기를 나누고 싶습니다.”

또한, 컬리에선 기술위원회가 열리는데요,  
기술위원회는 '왜 그리고 무엇을 위해' 탄생했는지 소개합니다.”

MARKET  
*Kurly*





첫번째 이야기-김민중 님

컬리에 합류하기 전, 저의 경험을 소개합니다.

**NAVER**



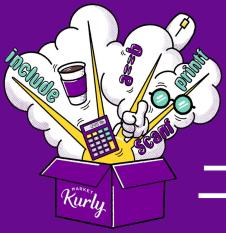
**LINE**



  
**bepro11**



MARKET  
*Kurly*



첫번째 이야기-김민중 님

그렇게 저는 컬리에 합류하게 되었습니다.



MARKET  
*Kurly*



첫번째 이야기-김민중 님

“컬리에 왜 합류하셨나요?” 라는 많은 질문에 대해 이렇게 답합니다.



내가 컬리에 합류한 가장 큰 이유?

= 커리어를 성장시키는 것

※ 커리어란? 내가 조직에서 어떤 부분을 기여했는지가 기록되는 것



커리어에서 무엇보다 중요한 것!

= 내가 했던 일 < 내가 한 일이 회사에 '어떻게' 기여했는가



= 내가 기여할 수 있는 부분이 정말 크겠군!!





첫번째 이야기-김민중 님

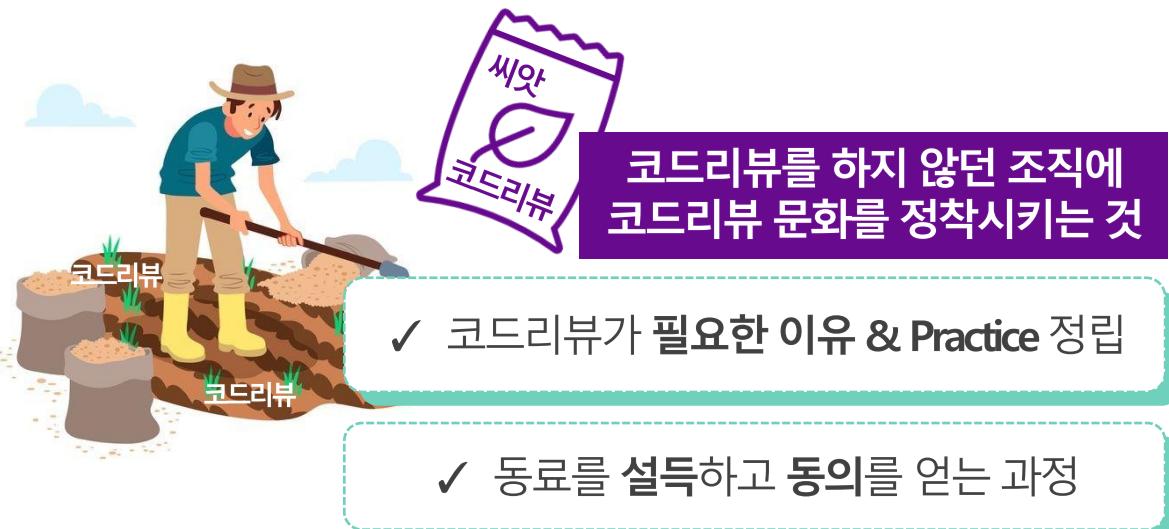
## '경험한 것'과 '경험을 만드는 것'의 차이



어느 쪽이 더 개발자로서 훨씬 높은 성장을 하게 될까요?



이미 정착된 코드리뷰 문화를 경험하는 것

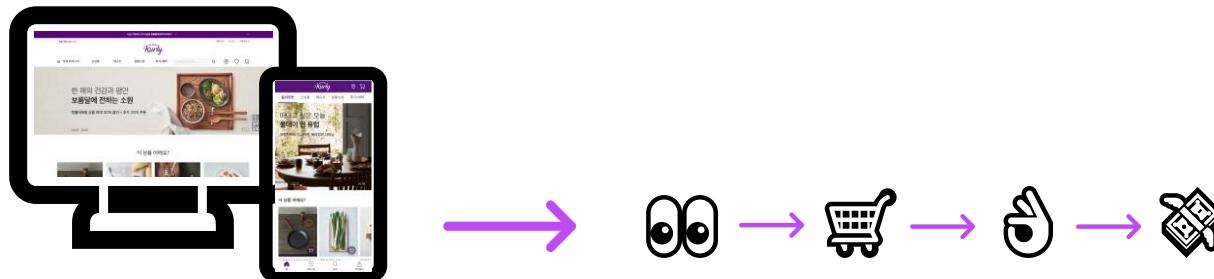




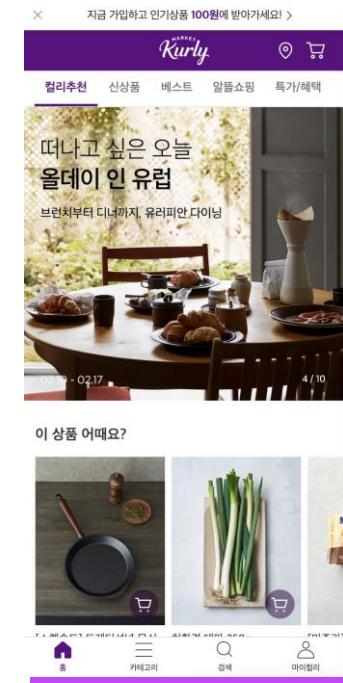
첫번째 이야기-김민중 님

제가 속한 커머스 플랫폼 – 웹 프론트 팀은,

### 커머스 플랫폼 – 웹 프론트 개발



PC, 모바일 웹 도메인에서  
고객분들이 마켓컬리 몰에서 물건을 구매하는 전 과정을 책임지는 팀입니다.





첫번째 이야기-김민중 님

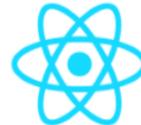
## 저희 팀의 테크 스택을 소개합니다.



웹 프론트엔드 팀에서는 아래의 테크 스택을 사용하고 있습니다.



Redux



ReactJS



Storybook



ReactiveX



NextJS



TypeScript



첫번째 이야기-김민중 님

컬리에서 여전히 더 도전하고 만들어 나가야 할 과제는 많습니다.

Kurly  
MARKET

Kurly  
MARKET

커머스 플랫폼 – 웹 프론트 개발

CHALLENGE  
CHALLENGE  
CHALLENGE  
CHALLENGE  
CHALLENGE



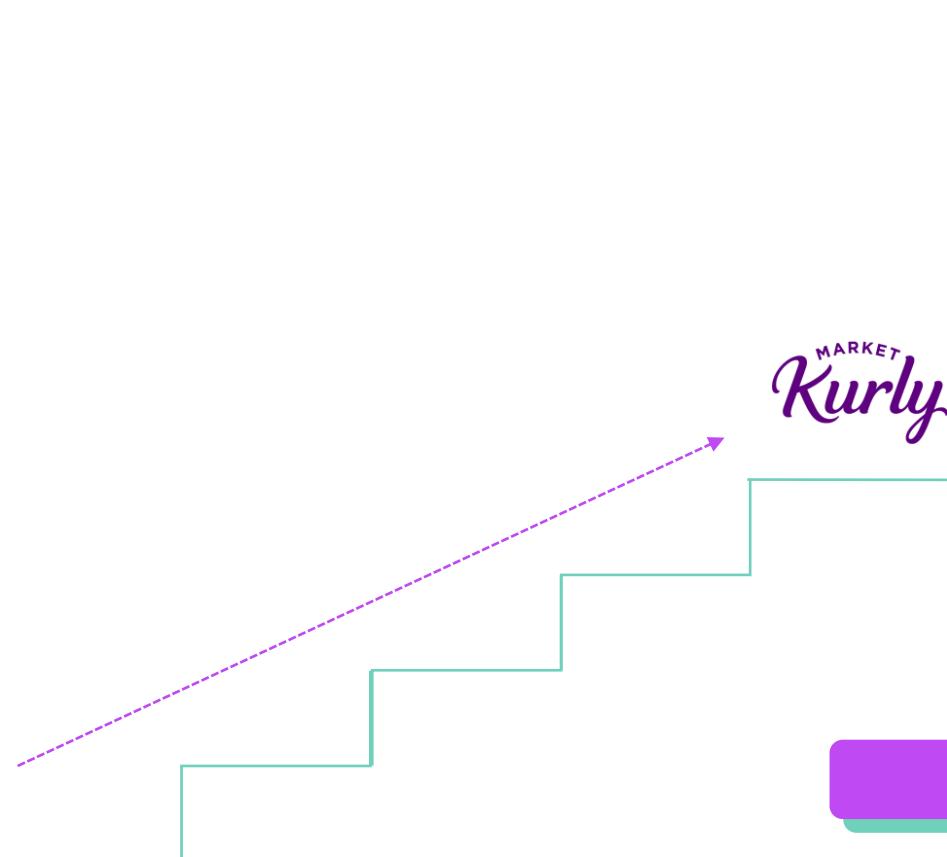
CHALLENGE

CHALLENGE

CHALLENGE

CHALLENGE

CHALLENGE





첫번째 이야기-김민중 님

저희 팀의 굵직굵직한 과제를 들여본다면,

레거시 → MSA

BFF 설계 및 구현

디자인 시스템 구축



그 외 비즈니스 요구사항을 효율적으로 처리하는 방법!

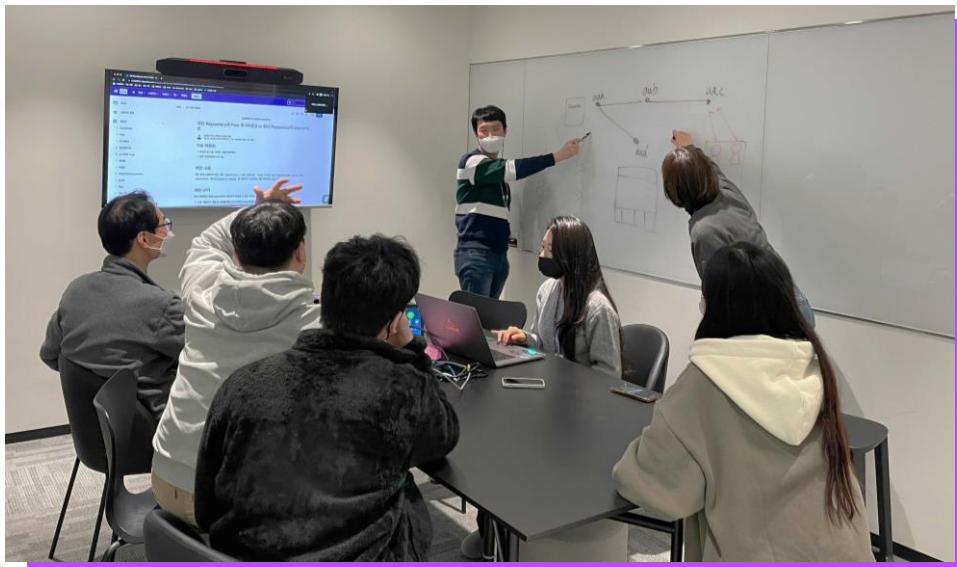


이 모든 과제들을 비즈니스 요구 사항을 구현해 나가면서 함께 성장시켜 나가야 하죠.



첫번째 이야기-김민중 님

## 기술위원회를 소개합니다!



언제 열리나요?

새로운 기술을 도입할 때

기존의 사용하던 기술을 새로운 기술로 대체할 때



어떻게 진행되나요?

기술에 관한  
간단 설명을  
문서로 팀에 제시  
**\*도입하고자 하는 이  
유 포함**

기술에 대한  
학습 시간 부여  
**\*1주일 이상**

기술의 도입 여부  
찬반 토론 진행  
**\*기술 도입에 대한 실  
과득을 비교 확인**

도입 여부  
최종 결정  
**\*명확한 근거를 기반**

➤ 자발적으로 근거를 만들고 그 근거에 따른 결정을 하게 되는 것이며 이러한 문화는 개발자 성장에 큰 도움이 되는 문화입니다.



첫번째 이야기-김민중 님

## 컬리에서 얻을 수 있는 높은 가치

컬리에서는 **높은 기여를 통한 빠른 성장을 할 수 있습니다.**

### Why?

구현이 필요한 과제 有  
설계 단계부터 참여 가능

신규 기술 도입에 대한 근거를  
만드는 과정을 통해 성장 가능

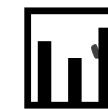


첫번째 이야기-김민중 님

저희는 이런 동료의 합류를 기다려요!



성장을 하고 싶은 개발자



객관화 근거를 기반으로  
합리적인 결정을  
팀에서 할 수 있도록 돋는 개발자



컬리 개발자상장기

## 두번째 이야기의 주인공

저는

“느리더라도 꾸준하게 성장 하고픈 거북이 같은 개발자”

방신철 입니다.



저는 조금 더 나은 세상을  
만들기 위한 서비스를 개발하는 것  
그리고 트렌드에 맞는  
새로운 기술을 적용하여  
개발하는 것을 좋아합니다.

> 방신철님 | 커머스플랫폼회원마케팅서비스개발

[컬리의 코프링 & 헥사고날 아키텍처 적용 사례]

“개발자란비즈니스생태계를 이해하고 발견된 문제점을  
가장 효율적인 기술로 해결하는 사람들이죠.”

컬리에서 스터디와 몸프로그래밍, 코드리뷰로 팀원들과 협업하여  
특정 비즈니스 문제점을 포착하고 개선했던 스페셜 경험을 공유합니다.”

MARKET  
*Kurly*





두번째 이야기 – 방신철 님

**SYSTEM: 문제가 발생했습니다.**



계정 찾기 인증 방식의 부족

※ 기존에는 이메일 인증 방식만 제공

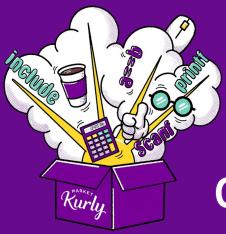
→ 컬리 가입 시 사용했던 이메일 주소를 잊었을 때 계정을 찾을 방법이 없음



수치화 된 값으로 문제를 정의해보자!

※ 월 약 4,300건 가량의 고객 불편 사항이 지속적으로 접수됨

→ 컬리 서비스 내에 전체 인입되는 고객 불편사항에서 4%의 큰 비중을 차지함



두번째 이야기 – 방신철 님

## 어떻게 해결해볼까? - 목표의 수립



다양한 인증 수단의 옵션 고려

→ 가장 대중적이고 익숙한 '**휴대폰 인증 방식**'을 추가 제공하자!



좀 더 욕심을 내볼까?

기존의 PHP 기반의 모놀리식 애플리케이션을 통해 제공되던 계정 찾기 기능은,  
다른 기능의 문제로 장애가 발생했을 경우에도 영향을 줄 수 있는 상황.

확장성, 장애 격리성을 보장할 수 있는 아키텍처의 새로운 마이크로 서비스로 개선해보자.

→ 기존의 모놀리식 애플리케이션에서 기능을 분리해보자!



두번째 이야기 – 방신철 님

저희는 이런 도구들을 사용합니다.



프로젝트를 진행하면서 기술적인 성장도 함께 챙기자!



기존 개발 스택



새로운 언어 채택



First Language로 코틀린 언어를 채택한 이유는?

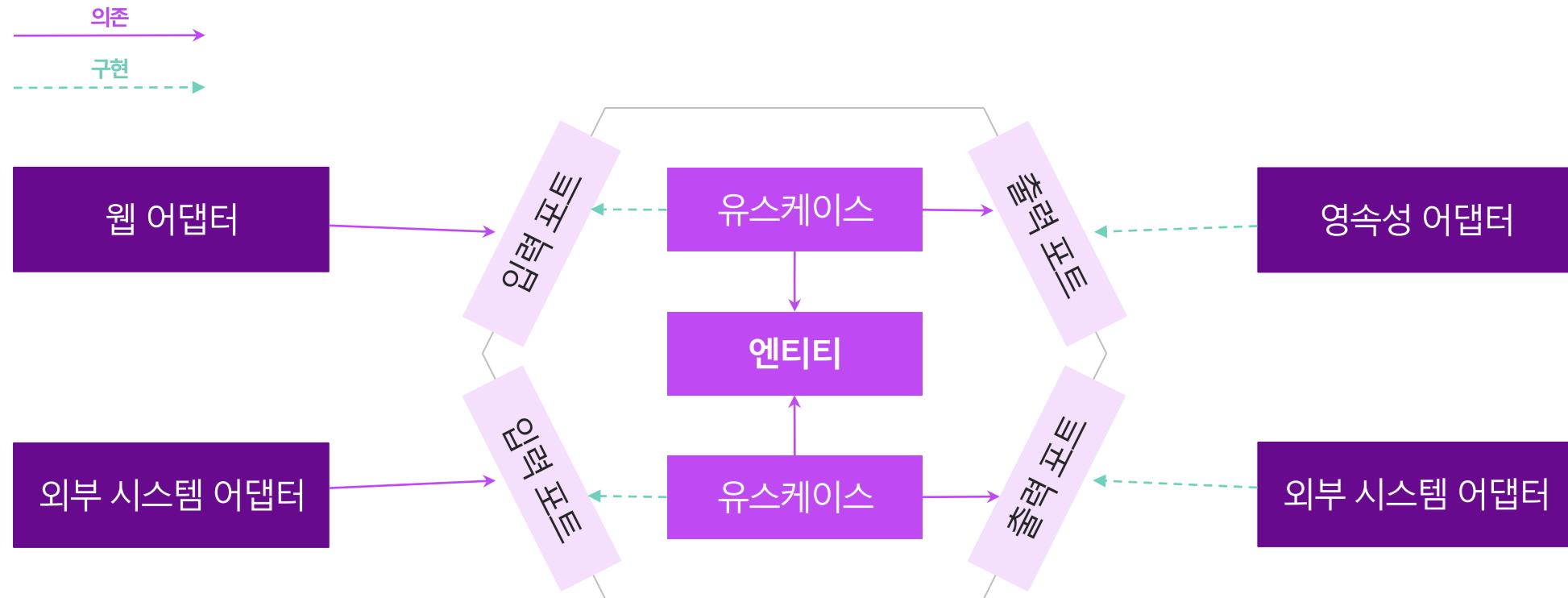
조금 더 가독성 좋은 코드를 작성, 유지보수하기 좋은 클린 아키텍처로 개발해보자.

팀 내의 자바 & PHP 개발자의 효율적인 co-work을 위해 채택!



두번째 이야기 – 방신철 님

## 헥사고날 아키텍처의 도입 배경



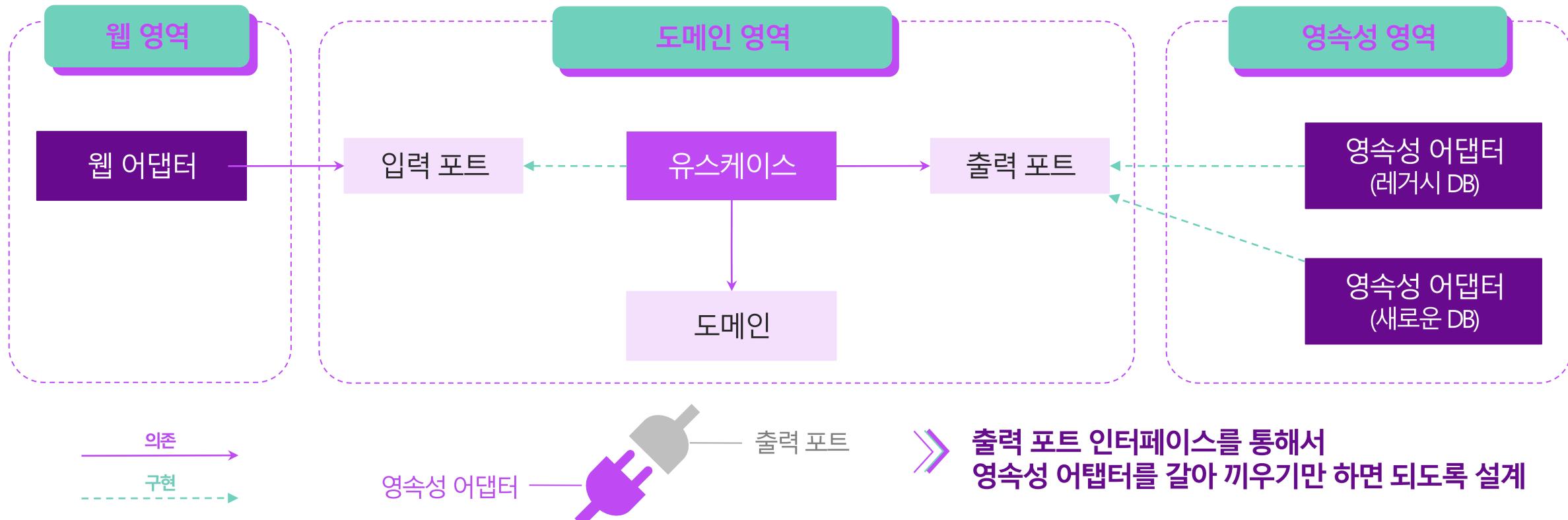


두번째 이야기 – 방신철 님

우리는 이렇게 적용해보았습니다!

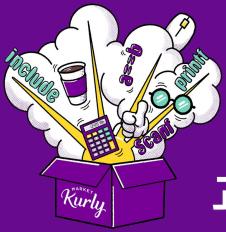


헥사고날 아키텍처 적용을 위한 멀티 모듈 구조 프로젝트 구성



Kurly  
MARKET

➤ 출력 포트 인터페이스를 통해서  
영속성 어댑터를 갈아 끼우기만 하면 되도록 설계



두번째 이야기 – 방신철 님

## 과정을 통해 느낀 어려움과 값진 경험



“이런 어려움을 겪었어요!”

- 러닝 커브
- 많은 포트 및 어댑터 생성 → 구조적으로 복잡



“하지만 이런 장점을 경험했어요!”

- 유스케이스별 깔끔한 분리 → 협업이 편리해짐
- 깨진 창문 이론 방지 → 깔끔한 객체지향 개발 가능



두번째 이야기 – 방신철 님

우리가 만들어 낸 결과는?



그래서 해당 프로젝트에서 어떤 결과를 얻으셨나요?

### 수량적 결과

- ✓ 월에 약 4300건 정도 발생하던 고객 불편사항을 9~60건으로 대폭 줄임.
- ✓ 확장성이 용이하고 장애 격리된 새로운 애플리케이션을 성공적으로 런칭함.

### 역량 성장 결과

- ✓ 팀 내에 코틀린+스프링부트 개발 스택을 잘 안착 시킬 수 있었음.
- ✓ 동료들의 기술적 역량 성장.



## 두번째 이야기 – 방신철 님

### 특별한 커리만의 협업 방식

#### 몹 프로그래밍

- 한명의 드라이버와 여러 명의 프로그래머가 하나의 PC로 코딩 또는 문서화 작업을 진행하는 방식으로 1:1 방식인 페어 프로그래밍을 1:N으로 확장시킨 형태의 개발 방식
- Covid19로 인한 재택근무 방식에 따라, IntelliJ IDEA의 Code With Me 활용하여 온라인 방식의 몹 프로그래밍 진행

#### 코드리뷰

- 서로의 코드에 대해 자유롭게 논의
- 단, 코드리뷰를 통해 PR Approve가 되어야 코드베이스에 머지 할 수 있도록 규칙을 생성
  - \*나의 동료가 어떤 왜 어떻게 작업했는가 파악
  - \*동료가 보지 못한 포인트 더블체크하여 버그를 최소화, 클린 코드로 개선 피드백
  - \*동료의 작업물을 통해 인사이트/개발지식을 얻도록



참여하는 모두가 오너십을 가지고 임하며 서로 어렵거나 모르는 부분에 대해 부끄러워 하지 않고 공유할 수 있는 분위기로 함께 스터디하면서 즐겁게 진행함:  
기술적 커뮤니케이션을 통해 기술적 지식과 경험의 전달이 빠르고 효과적으로 이루어짐



## 두번째 이야기 – 방신철 님 건강한 코드리뷰 문화

### PR 포맷

BangShinChul commented on 9 Dec 2021 · edited

**개요** 🌟

카카오 API 서버로 카카오톡 채널 관계를 조회하는 Port 및 Adapter를 작성하였습니다.

추후에 해당 Port 및 Adapter를 사용하여

카카오톡 연동/회원가입 할 때 카카오톡 채널 관계 확인 API 를 호출해서 정보를 DB에 저장 하는 기능을 작성할 예정입니다.

**작업 사항** ━

- 카카오톡 채널 관계 enum 클래스를 작성하였습니다.
- 카카오톡 채널 관계 조회 Port 및 Adapter를 작성하였습니다.
- 카카오톡 채널 관계 조회 Adapter의 테스트 코드를 작성하였습니다.

**참고 문서** ━

- 카카오톡 채널 관계 확인하기 API 문서 링크
- Spring Cloud Config : 카카오톡 채널 uuid 프로퍼티 참고

**체크 리스트** ✓

PR을 발행하기 이전에 아래의 항목을 확인해 주세요.

작성된 코드에 대해 테스트 코드를 적절히 추가, 변경, 삭제 하였습니까?

✓ Test Results

com.kurly.member.auth.adapter.social.kakao.FindKakaoChannelRelationAdapterTest 797 ms

- ✓ find 메서드는
  - ✓ 정상적인 카카오 채널 관계 정보를 응답받을 경우 382 ms
  - ✓ 정상적인 카카오 채널 관계 정보를 리턴한다.
- ✓ 빈 카카오톡 채널 관계 정보를 응답받을 경우 9 ms
- ✓ null을 리턴한다.
- ✓ 기존에 정의되지 않은 카카오톡 채널 관계 정보를 응답받을 경우 9 ms
- ✓ 예외를 리턴한다.

모든 테스트 코드가 성공 하였습니까? (명령어 : ./gradlew test)

### 실제 코드리뷰 내용

common/src/main/kotlin/com/kurly/member/auth/exception/ErrorCodes.kt Outdated

```

37 +     * @param httpStatusCode Http Status Code
38 +     * @return 공통 오류 코드 객체 code 값
39 +     */
40 +    fun getCode(httpStatusCode: Int): String {

```

binary-park on 3 Sep 2021

of 메시드 명을 사용하면 어울까요??

BangShinChul on 3 Sep 2021

@binary-park 리뷰 감사합니다 원영님!

of 메시드 명이 더 가독성이 좋을까요..?

binary-park on 3 Sep 2021 · edited

조슈아님이 이해하기 위해서 일반적인 텍토리 메시드 양을 소개 할때  
of, from, getInstance 등을 소개 해주고 있어서요 ㅎㅎ

친 후로 아래와 같이 구분해서 사용 합니다.  
주입된 것과 primitive 타입인 of을  
주입된 것과 객체라면 from을 사용합니다.  
싱글톤이라면 getInstance

BangShinChul on 3 Sep 2021

앞 키워드 조수이정..  
감사합니다 원영님! 개선해서 다시 전달드리겠습니다!

BangShinChul on 6 Sep 2021

@binary-park 기준 getCode 메서드를 of 메서드로 명칭을 수정하였습니다.

```

+ companion object {
+     /**
+      * Http Status Code에 맞은 공통 오류 코드 객체 code 값을 리턴.
+      *
+      * @param httpStatusCode Http Status Code
+      * @return 공통 오류 코드 객체 code 값
+      */
+     fun of(httpStatusCode: Int): String {
+         return when (httpStatusCode) {
+             400 -> BAD_REQUEST.code
+             401 -> UNAUTHORIZED.code
+             403 -> FORBIDDEN.code
+             404 -> ENTITY_NOT_FOUND.code
+             406 -> NOT_ACCEPTABLE.code
+             415 -> UNSUPPORTED_MEDIA_TYPE.code
+             422 -> INVALID_INPUT_VALUE.code
+             500 -> INTERNAL_SERVER_ERROR.code
+             504 -> GATEWAY_TIMEOUT.code
+             else -> INTERNAL_SERVER_ERROR.code
+         }
+     }
+ }

```

binary-park on 9 Sep 2021

로그에 대한 궁금증...

- 해당 로그가 실제 필요하지?
- 해당 로그가 상세히 info 인지 debug인지
- 해당 로그가 보안정책에 위배되지 않는지... (개인정보를 로그에 적으면 안됨, 마스킹 여부에 따라 다른지는 모르겠네요.)

binary-park on 9 Sep 2021 · edited

자 이제 코드를 작성해 보자면 조금 더 가독성 좋은 코드를 작성해 봅시다~

어제의 구조는 find 메서드에서 의사코드 느낌으로 수행되는 절차들을 애서드로 뛰어서 이렇게 코드를 하게되면 절차가 명확하게 보이는 점이 있습니다. 아래의 코드 예를 들어보면:

- 회원 오류를 조회하는 코드만 오류를 발생한다.
- 첫 번째 회원을 기준으로 토큰을 생성한다.
- 이메일을 발행한다.

처음 요구사항이 명확하게 보이고, 해당 절차의 상세 구현을 애서드 내부에 정의되어 됩니다. 코드를 전부 보자 일睹하면 다른 팀이 해당 유스케이스를 쉽게 이해할 수 있겠죠? 상상해보니...

서비스 단위로 영역의 구분이 되어 있기 때문에 아래에 발송이 잘 안된다면 sendEmail() 대신에 해당 클래스의 예외/해임 커린다면 서비스가 플러스로 도출될 수도 있겠죠?

BangShinChul on 9 Sep 2021

@binary-park 리뷰 감사합니다 원영님!

피드백 주신대로 개선해보도록 하겠습니다.ㅎㅎ

```

+ override fun find(command: FindPasswordByEmailCommand) {
+     val members = findMembersIfEmptyThrowException(command)
+     val (member, token) = createTokenByFirstMember(members)
+     sendEmail(command, member, token)
+ }

```

BangShinChul on 9 Sep 2021

@binary-park 와... 정말 많이 배웁니다..! 👍 감사합니다 원영님!

피드백 주신대로 개선해보도록 하겠습니다!

아, 그리고 로그에 대해서는 제가 생각이 짧았던 것 같습니다.

일단 비밀번호 재설정 링크 이메일 발송의 경우, 허나의 계정에만 전송해야하는데  
즉, 회원 정보 조회 시 memberId라는 유니크 값으로 조회하는데 만약 여러 회원 정보가 리턴된다면, 웬지 과학하고 있어야 할 것 같아서 로그를 찍어두면 좋겠다 싶어서 작성했던 코드였습니다.

이 부분에 대해서는 추후에 저희끼리 내부적으로 논의한 후에(로그를 찍어야 하는지, 로그를 찍어야 한다면 어떤 구조로 출력해야 하는지 등) 작성하는것이 좋을 듯 싶습니다! 따라서 해당 로그 출력 코드는 제거하도록 하겠습니다!



두번째 이야기 – 방신철 님

저희는 이런 동료의 합류를 기다려요!



- ✓ 본인이 무엇을 모르는지 알고, 이를 학습해서 기술적 성장을 하시는 분



- ✓ 모르는 것을 부끄러워 하기 보다는 더 드러내어 성장하기 위해 노력하시는 분



- ✓ 학습한 기술이나 경험을 동료에서 전파하여 기술적 상향 평준화를 이루고자 하시는 분



컬리 개발자성장기

## 세번째 이야기의 주인공

저는

“양질의 코드와 지속 가능한 프로덕트를 고민하고  
비즈니스를 코딩과 같은 열정으로 이해하고자 노력하는 평범한 개발자”

최준영 입니다.



> 최준영님 | 풀필먼트 프로덕트 출고서비스개발

[더 나은 출고 서비스를 위한 항해 일지]

“컬리에서 안정적인 출고 서비스를 제공하기 위해 겪었던  
흥미로운 항해일지와 앞으로의 방향성을 공개합니다!

더불어 팀원들과 어떤 방법으로 논의하며 함께 성장할 수 있었는지,  
저희만의 커뮤니케이션 문화에 대해서도 소개합니다.”

MARKET  
*Kurly*



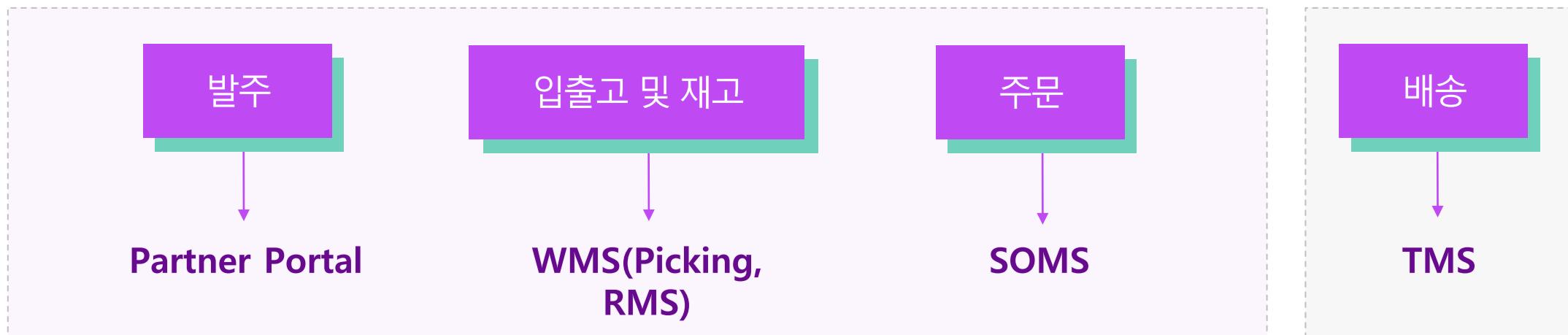


세번째 이야기 – 최준영 님

## 풀필먼트 소개

### 컬리 풀필먼트

### 컬리 딜리버리





세번째 이야기 – 최준영 님

## 풀필먼트 소개



피킹(Picking)이란?



### 피킹 (picking)

[명사] [경제] 출고할 상품을 물류 창고의 보관 장소에서 꺼내는 일



세번째 이야기 – 최준영 님

## 풀필먼트 소개



최적화란?



**최적화**

» 주문을 상품으로 그룹화하여, 작업자가 최적의 동선으로  
더 빠른 피킹을 가능하게 데이터를 가공하는 방법



세번째 이야기 – 최준영 님

## 컬리 풀필먼트(물류) 센터

송파, 화도 물류센터



DAS

\*DAS(Digital Assorting System):

회차별 각 오더유형별 주문 품목을 한번에 피킹(총괄 피킹)하고 피킹된 상품은 DAS존에 비치된 빈 바구니에 각 고객별 주문상품을 재분류하는 시스템

김포 물류센터



QPS, DAS

\*QPS(Quick Picking System):

상품 분리 담당자가 레일을 통해 자신 앞으로 이동해 온 상품을 시스템 지시에 따라 상자에 담고 바로 이어 포장 단계로 넘길 수 있도록 설계되어 생산성 향상 및 작업 환경을 개선한 자동화 시스템



세번째 이야기 – 최준영 님

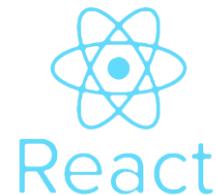
## 컬리 김포CC PDA Picking 기술 스택



김포 PDA Picking에 사용된 모든 기술 스택을 소개합니다!



android



BABEL



webpack



Jest

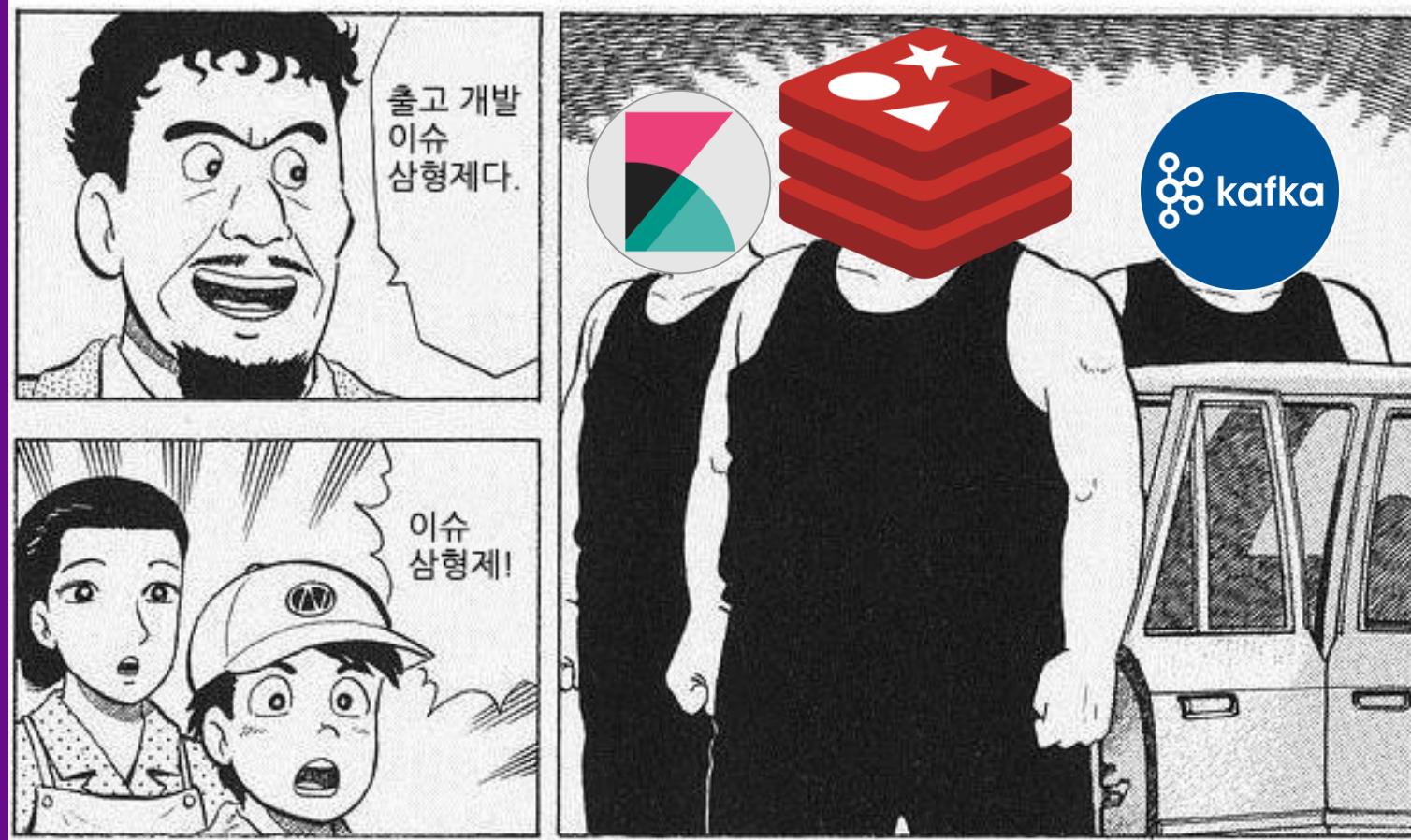


MARKET  
Kurly



세번째 이야기 – 최준영 님

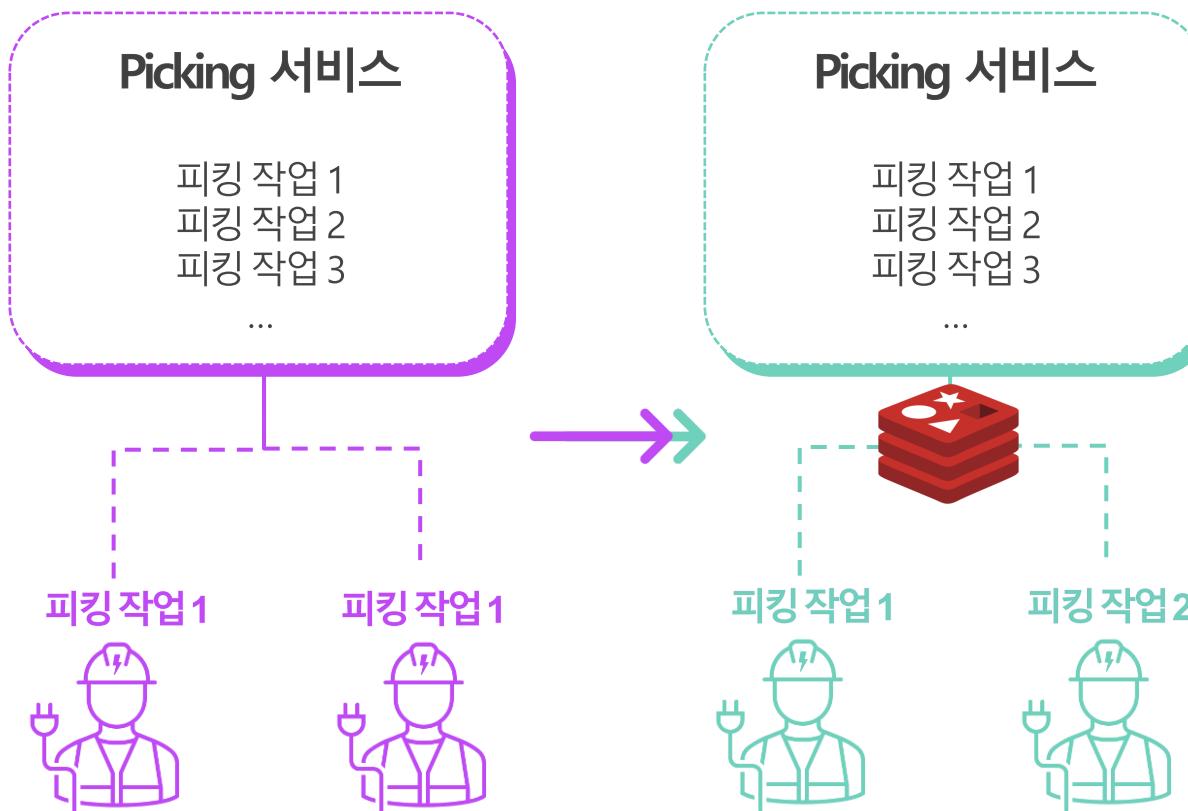
## 개발 이슈에 대해 소개합니다





세번째 이야기 – 최준영 님

## 이슈1 (동시성 제어, Redis Distributed Lock)



```

protected void initRedisDistributedLock() {
    this.redissonClient = SpringUtils.getBean(name: "redisson");
    this.lockKey = lockKey + this.distributedKey;
    this.redisDistributedLock();
}

protected void redisDistributedLock() {
    log.info("redisDistributedLock !! {}", this.lockKey);
    this.rLock = redissonClient.getLock(this.lockKey);
    this.rLock.lock(leaseTime: 10, TimeUnit.SECONDS);
}

protected void redisDistributedUnLock() {
    log.info("redisDistributedUnLock !! {}", this.lockKey);
    this.rLock.unlock();
}

redis> keys PDA-CC02:picking_batch_last_qps:*
qps:*
1) "PDA-CC02:picking_batch_last_qps:GGM1"
2) "PDA-CC02:picking_batch_last_qps:GGH1"

```



## 세번째 이야기 – 최준영 님 이슈2 (로그 확인, Kibana)

**DATADOG**

Traces > logistics-outbound-picking-cc02 > POST /graphql > trace\_id: 5540766736139344252

Feb 10 19:06:04.962 | 50.8 ms p74 ▲ POST http://picking-cc02.logistics.kurly.services/graphql 200 OK

Flame Graph Span List Trace Map

servletrrequest POST /graphql

redis... redis.qu... oracle.query select a... oracle.query select... oracle.query select... oracle.query update... repository.operation PickingTaskRepo... oracle.query se...

4.49 ms

Logs 9

Tags Infrastructure Metrics Logs 9 Processes Network Code Hotspots

Trace ID trace\_id:5540766736139344252

DATE SERVICE HOST

Feb 10 19:06:04.965 logistics-outbound-picking-cc02 logistics-outbound-picking-cc02-AS-152-9

=====> USER : GP043 REQUEST : com.kurly.lpd.outboundpicking.core.gimpo.adapter.in.PickingTaskMutationResolver(mark...  
totelId=T100016079, picker=GP043, warehouseKey=GM1, documentDate=2022-02-10, requestShippingDate=2022-02-11, processType=MULTI...  
0079, skuCode=MK00000097524, skuAlternativeCode=8801115135184, skuName=[서울우유] 맬기 우유 280ml 3개입, skuSubstituteName=(없음) [A...  
Feb 10 19:06:04.967 logistics-outbound-picking-cc02 logistics-outbound-picking-cc02-AS-152-9

=> Preparing: select /\*+ index(a PICKING\_MASTER\_IDX\_08) \*/ a.PICKING\_MASTER\_ID AS pickingMasterId ,a.WAREHOUSE\_KEY AS wareh...  
,a.PROCESS\_SUB\_TYPE AS processSubType ,a.GROUP\_NO AS groupNumber ,a.WORK\_BATCH\_NO AS workBatchNumber ,a.PICKING\_BATCH\_NO AS pa...  
,a.SKU\_SUB\_NAME AS skuSubstituteName ,a.PICKING\_ZONE\_ID AS pickingZoneId ,a.LOCATION\_KEY AS locationKey ,a.LOCATION\_NAME AS locat...  
Feb 10 19:06:04.968 logistics-outbound-picking-cc02 logistics-outbound-picking-cc02-AS-152-9

=> Parameters: GM-092381-002-0079(String), GM-092381-002(String), GM-092381(String)

Feb 10 19:06:04.972 logistics-outbound-picking-cc02 logistics-outbound-picking-cc02-AS-152-9

**KIBANA**

pickingBatchNumber : 'GL-091883-001-0012'

+ Add filter

prod-log\_application-outbo... 49 hits

Search field names

Filter by type 0

Selected fields 4

picker  
actionType  
keyCode  
resultStatus

Available fields 86

Popular  
@timestamp  
actionTime  
pickingBatchNumber  
resultJson.problemPickingQuantity  
resultJson.skuCode  
resultJson.skuName  
resultJson.totl  
resultMessage  
\_id

Time picker

Time ↓ picker

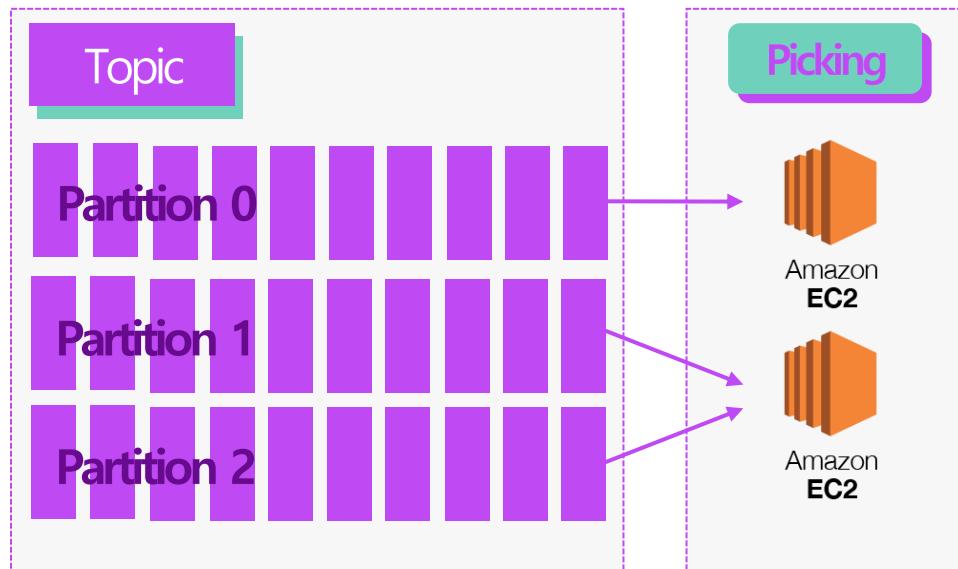
Time	picker	actionType	keyCode
> Feb 10, 2022 @ 19:25:41.645	GP640	BATCH_COMPLETE	GL-092385-001-0014
> Feb 10, 2022 @ 19:25:41.510	GP640	SKU_SCAN	8809660652350
> Feb 10, 2022 @ 19:25:41.048	GP640	SKU_SCAN	8809660652350
> Feb 10, 2022 @ 19:25:33.264	GP640	LOCATION_SCAN	13SC0209103
> Feb 10, 2022 @ 19:24:56.353	GP640	SKU_SCAN	8801114149355
> Feb 10, 2022 @ 19:24:51.822	GP640	LOCATION_SCAN	13SC0209101
> Feb 10, 2022 @ 19:24:20.735	GP640	SKU_SCAN	8809663130664
> Feb 10, 2022 @ 19:24:20.067	GP640	SKU_SCAN	8809663130664
> Feb 10, 2022 @ 19:24:08.256	GP640	LOCATION_SCAN	13SC0101203
> Feb 10, 2022 @ 19:23:44.829	GP640	SKU_SCAN	8809536620230



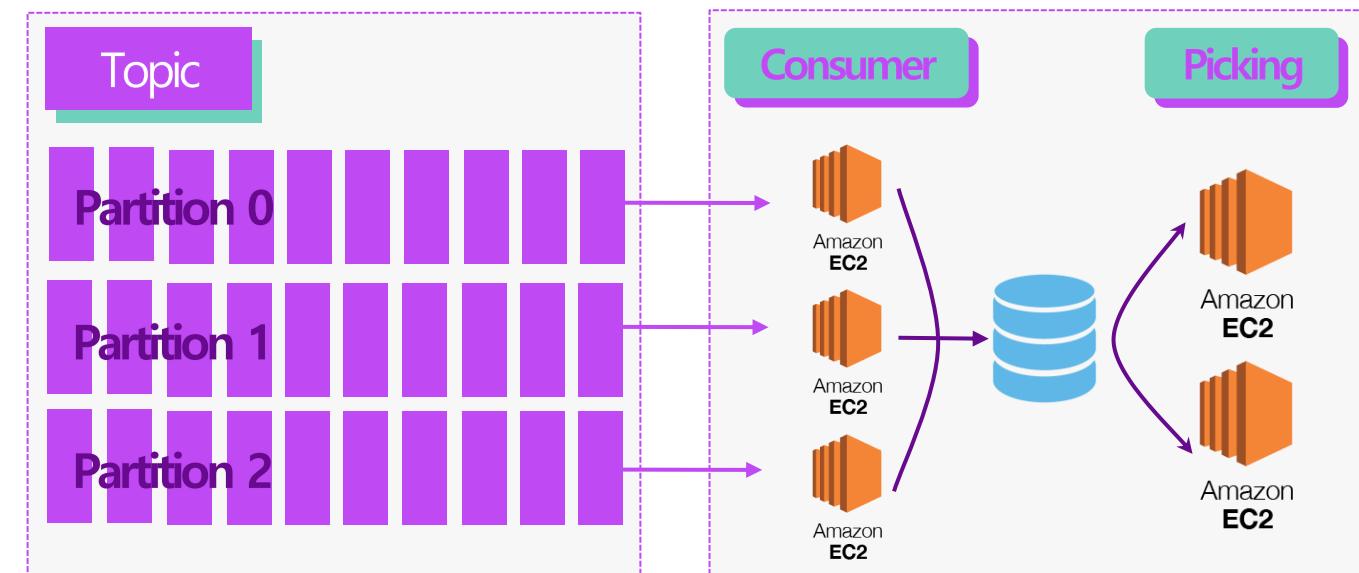
세번째 이야기 – 최준영 님

## 이슈3 (컨슈머 분리, Kafka Consumer)

As-Is



To-Be



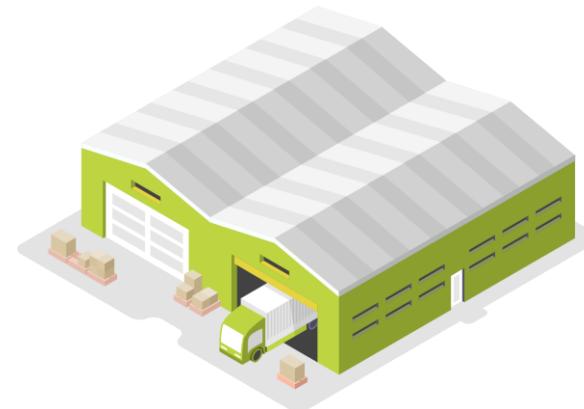


세번째 이야기 – 최준영 님

## 앞으로의 방향성



출처 : <https://www.youtube.com/watch?v=YtztxnXeAb4>

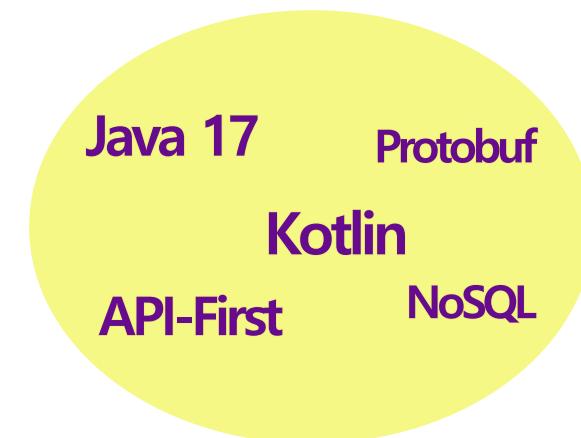


### 김포 DAS 도입

\*DAS(Digital Assorting System):  
회차별 각 오더유형별 주문 품목을 한번에 피킹(총괄 피킹)하고 피  
킹된 상품은 DAS존에 비치된 빈 바구니에 각 고객별 주문상품을  
재분류하는 시스템

### 신규 센터 오픈

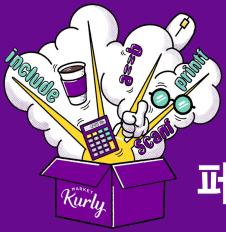
### 신규 기술 도입





세번째 이야기 – 최준영 님  
팀 문화를 소개합니다.





세번째 이야기-최준영 님

# 페어 프로그래밍과 Code With Me를 소개합니다.



저희 팀의 특별한 개발 문화를 소개합니다.



```
main java com kurly lpd outboundpicking core gimpo adapter in mapper PickingTaskDtoMapper.java
src main
└── logistics-outbound-picking-cc02
    ├── build.gradle
    ├── gradle
    ├── gradle-wrapper.properties
    ├── LOGS_ABSOLUTE_PATH_IS_UNDEFINED
    └── main
        ├── com.kurly.lpd.outboundpicking
        │   ├── common
        │   ├── config
        │   ├── core
        │   │   ├── domain
        │   │   ├── infrastructure
        │   │   ├── mapper
        │   │   └── test
        │   └── application
        ├── resources
        ├── test
        └── Dockerfile
        └── entrypoint.sh
└── env
└── gradlew
└── gradlew.bat
└── gradlew.sh
└── README.md
└── 의부 라이선스
└── 스크래치 및 콘솔

Connected to hongjaesong's logistics-outbound-picking-cc02 - PickingTaskDtoMapper.java
File Project Tools Help
Code With Me: Port Forwarding Voice Call
hongjae... 3
Line Number Bar
1  adapter.in.mapper;
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18 kingTaskDto pickingTaskDto);
19 batchInfoByDto -> mapToTransferTotalInfoByDto
20
21     @Override
22     public void transferSkuInfoByDto(TransferPickingBatchInfoDto transferPickingBatchInfoDto,
23                                     List<TransferskuInfoDto> transferSkuInfoDtos);
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
99
100
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
117
118
119
119
120
121
122
123
124
125
126
127
127
128
129
129
130
131
132
133
134
135
135
136
137
137
138
139
139
140
141
142
142
143
144
144
145
145
146
146
147
147
148
148
149
149
150
150
151
151
152
152
153
153
154
154
155
155
156
156
157
157
158
158
159
159
160
160
161
161
162
162
163
163
164
164
165
165
166
166
167
167
168
168
169
169
170
170
171
171
172
172
173
173
174
174
175
175
176
176
177
177
178
178
179
179
180
180
181
181
182
182
183
183
184
184
185
185
186
186
187
187
188
188
189
189
190
190
191
191
192
192
193
193
194
194
195
195
196
196
197
197
198
198
199
199
200
200
201
201
202
202
203
203
204
204
205
205
206
206
207
207
208
208
209
209
210
210
211
211
212
212
213
213
214
214
215
215
216
216
217
217
218
218
219
219
220
220
221
221
222
222
223
223
224
224
225
225
226
226
227
227
228
228
229
229
230
230
231
231
232
232
233
233
234
234
235
235
236
236
237
237
238
238
239
239
240
240
241
241
242
242
243
243
244
244
245
245
246
246
247
247
248
248
249
249
250
250
251
251
252
252
253
253
254
254
255
255
256
256
257
257
258
258
259
259
260
260
261
261
262
262
263
263
264
264
265
265
266
266
267
267
268
268
269
269
270
270
271
271
272
272
273
273
274
274
275
275
276
276
277
277
278
278
279
279
280
280
281
281
282
282
283
283
284
284
285
285
286
286
287
287
288
288
289
289
290
290
291
291
292
292
293
293
294
294
295
295
296
296
297
297
298
298
299
299
300
300
301
301
302
302
303
303
304
304
305
305
306
306
307
307
308
308
309
309
310
310
311
311
312
312
313
313
314
314
315
315
316
316
317
317
318
318
319
319
320
320
321
321
322
322
323
323
324
324
325
325
326
326
327
327
328
328
329
329
330
330
331
331
332
332
333
333
334
334
335
335
336
336
337
337
338
338
339
339
340
340
341
341
342
342
343
343
344
344
345
345
346
346
347
347
348
348
349
349
350
350
351
351
352
352
353
353
354
354
355
355
356
356
357
357
358
358
359
359
360
360
361
361
362
362
363
363
364
364
365
365
366
366
367
367
368
368
369
369
370
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
401
401
402
402
403
403
404
404
405
405
406
406
407
407
408
408
409
409
410
410
411
411
412
412
413
413
414
414
415
415
416
416
417
417
418
418
419
419
420
420
421
421
422
422
423
423
424
424
425
425
426
426
427
427
428
428
429
429
430
430
431
431
432
432
433
433
434
434
435
435
436
436
437
437
438
438
439
439
440
440
441
441
442
442
443
443
444
444
445
445
446
446
447
447
448
448
449
449
450
450
451
451
452
452
453
453
454
454
455
455
456
456
457
457
458
458
459
459
460
460
461
461
462
462
463
463
464
464
465
465
466
466
467
467
468
468
469
469
470
470
471
471
472
472
473
473
474
474
475
475
476
476
477
477
478
478
479
479
480
480
481
481
482
482
483
483
484
484
485
485
486
486
487
487
488
488
489
489
490
490
491
491
492
492
493
493
494
494
495
495
496
496
497
497
498
498
499
499
500
500
501
501
502
502
503
503
504
504
505
505
506
506
507
507
508
508
509
509
510
510
511
511
512
512
513
513
514
514
515
515
516
516
517
517
518
518
519
519
520
520
521
521
522
522
523
523
524
524
525
525
526
526
527
527
528
528
529
529
530
530
531
531
532
532
533
533
534
534
535
535
536
536
537
537
538
538
539
539
540
540
541
541
542
542
543
543
544
544
545
545
546
546
547
547
548
548
549
549
550
550
551
551
552
552
553
553
554
554
555
555
556
556
557
557
558
558
559
559
560
560
561
561
562
562
563
563
564
564
565
565
566
566
567
567
568
568
569
569
570
570
571
571
572
572
573
573
574
574
575
575
576
576
577
577
578
578
579
579
580
580
581
581
582
582
583
583
584
584
585
585
586
586
587
587
588
588
589
589
590
590
591
591
592
592
593
593
594
594
595
595
596
596
597
597
598
598
599
599
600
600
601
601
602
602
603
603
604
604
605
605
606
606
607
607
608
608
609
609
610
610
611
611
612
612
613
613
614
614
615
615
616
616
617
617
618
618
619
619
620
620
621
621
622
622
623
623
624
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
145
```

페어 프로그래밍

# Code With Me

- ✓ 서로의 지식을 공유하며 개발
  - ✓ 같은 목표에 집중하여 더 빠른 결정
  - ✓ 실시간 코드 리뷰 가능
  - ✓ 드라이버는 작은 목표 네비게이터는 더 넓은 시야



## 세번째 이야기 – 최준영 님 저희 팀은 문서에 진심입니다.



저희 팀의 특별한 개발 문화를 소개합니다.

# 진지 문서화

**기본 전략**

앞서 말했듯 모든 배포는 QA 검수가 이뤄져야 배포가 가능하게끔 구조 변경을 해야했다.  
그리므로 티켓이 발생될 때마다 개발자는 **feature branch** 를 생성하고, 개발 완료 시 QA 팀과 협동을 진행하여 **QA branch** 를 생성한다. 만약 개발 범위가 비슷하지만 티켓이 나뉘어져 있다면 QA와 조율하여 한 **QA branch** 에 여러 **feature branch** 를 병합하여 QA를 진행한다.

- **master**: 현재 운영 Product에 배포된 내용과 동일한 내용의 브ran치
- **stage**: 차기 운영 Product에 배포되기 전으로, 당장 내일 배포가 되어도 문제가 없는 브란치
- **qa**: QA 팀수를 받기 위한 브란치
- **feature**: 신규 개발을 위한 브란치

1. Jira에 티켓이 생성되면, 해당 내용에 대한 **feature branch** 를 생성한다  
2. 개발 완료 후 QA 일정이 정정되면 **QA branch** 를 생성하고 소스를 병합한다  
3. QA 검수가 완료될 경우, 개발자들은 **QA branch**에서 각자 개발한 소스를 병합한다  
4. 만약 동시에 다른 QA가 진행될 경우, 새로운 **QA branch** 를 생성하고 2~3번과 동일한 형태로 진행한다  
5. QA가 완료될 경우 **stage branch**에 소스를 병합한다  
6. **stage branch** 브란치에서 **master branch** 브란치로 소스를 병합하고 이미지를 생성하여 운영 Product에 배포한다

**개발자간 지켜야 할 규칙 - 이를**

이로서 branch 규칙은 어느 정도 정리가 되지만, 협업하기엔 아직 몇 가지 정리하고 넣어야 할 내용들이 있다.  
대표적으로 branch 명명 규칙, PR 규칙에 대해 다뤄보자 한다.

**1. branch명**

branch type이 Jira 티켓명으로 branch명을 생성하다보니, 해당 branch에 대한 내용을 확인하려면 Jira에 접속하여 해당 티켓 번호를 입력하고 조회해야 했다. 그래서 간략한 요약을 작성하여 한눈에 branch 내용을 파악하기 용이하게 작성하기로 했다.

**Outbound-스쿼드B**

당당 시스템

- vpn 영화박 혜재 요청
- 용어사전
- 개발 배포 매뉴얼
- 스크립트 배포 매뉴얼
- 운영 배포 매뉴얼

로그 확인 매뉴얼

Github Repository

KafkaDrop endpoint

주요 기술

기술 활동

- Server Architecture
- 로그 전략
- 코딩 컨벤션 IntelliJ 기본 설정
- Kibana 기초 사용법
- Insomnia
- Git Blame Author Name 설정

Team 문화, 규칙

- 그라운드 룰
- 코드 리뷰
- IntelliJ pair programming
- Git 사용 전략

회의록

- 대필리 미팅
- 김포 관리자 서비스 API 개선
- 버전스냅
- 기술블로그 기고?
- 신규 멤버 이소 코요

보관된 페이지

**Market Kurly 출고서비스개발 Squad B**

무슨 일을 하나요?

저희 팀은 마켓리에서 발생하는 물류의 출고 서비스 중 물류 퍼킹 시스템과 신규 WMS 고도화를 담당합니다. 고객님께서 주문하신 물건을 최대한 빠르고 정확하게 퍼킹할 수 있게 자동화하고, 현 WMS 시스템의 3.0 버전을 위한 신규 기술 도입을 연구하고 있습니다. 구체적으로는 다음과 같은 일정입니다.

- 하루 수십만건의 주문에서 더 빠르게 물품을 담을 수 있도록 최적화된 데이터로 현장 작업자가 PDA로 작업할 수 있도록 개발하고 운영합니다.
- 현장 관리자 관리, 차량, 통계, 모니터링할 수 있는 어드민 시스템을 개발하여 사용합니다.
- 김포 QPS 운영을 담당하는 WCS와 kafka를 연계하여 피킹 정보를 송수신합니다.
- 팀원들이 신규 WMS 시스템에 필요한 기술들을 각자 유기적으로 리서치하고 스터디하여 다가을 개발에 대비합니다.
- 개발자들은 두명의 페어를 이루는 페어 프로그래밍을 추구하여 더욱 건고한 시스템을 완성할 수 있도록 노력합니다.
- 모든 코드는 코드리뷰를 진행하고, 코드리뷰가 끝나면 QA를 시작합니다.
- 프로젝트가 시작되면 기획/운영/개발이 한팀으로 풍자 기관하게 움직입니다.
- 팀원 전체가 문서의 중요성을 알고 대부분의 내용을 문서화하여 기록합니다.

진행 사항

- 장지피킹부
- 김포 PDA 고도화 논의 시작

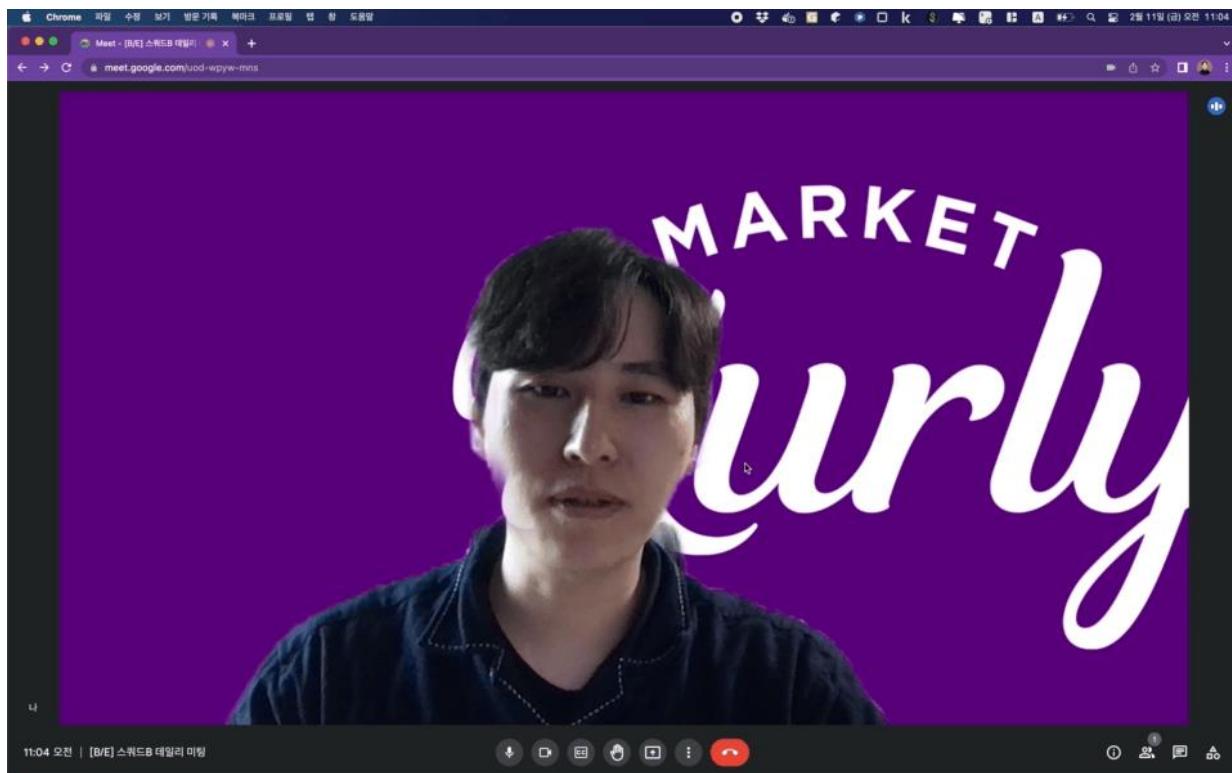




## 세번째 이야기 – 최준영 님 '버스터 콜' 들어보셨나요?



저희 팀의 특별한 개발 문화를 소개합니다.



※보안정보 및 개인정보 입력 금지※

### 2021.11.10 호출자 [오대호]

LOD우민식(이)가 만들 LOD오대호(이) 11월 11, 2021에 마지막 업데이트 · 이 페이지를 조회한 사람 11명

2021.11.10 버스터콜  
@LOD오대호 님의 호출

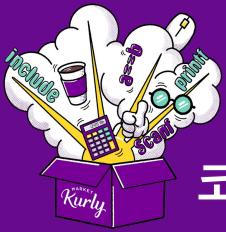
참석자

- @LOD홍재성
- @LOD최준영**
- @LOD최우림
- @LOD우민식

주제 : 김포 리펙토링 스팩을 어떻게 할까요?

### 결론

- 변경점이 있을경우 본인의 스크프 내에서 변경을 한다.
  - 전체 코드 레벨에서 변경이 필요한부분은 TODO로 코멘트를 남겨두고 코드리뷰시 팀원이 해당 TODO에 따봉을 달아서 인지하도록 한다.
  - 현재 일정내 가능하면 진행하되, 불가하다면 Todo로 남겨둔다.
- mybatis → jdbcTemplate 전환하지 않는다. 통계, 배치 쿼리 외 비즈니스 로직은 가능한 jpa로 이관한다.
  - 추가피킹 지시
- 추후 개선 과제는 TODO에 남기도록 한다. [관리자 API 서비스 분리](#)
- 유효성 검사는 애리포맷 일관화와 추후 graphql deprecated 를 고려하여 graphql 스키마 에러를 활용하지 않고 spring, java code 레벨에서 처리한다.



## 세번째 이야기 – 최준영 님 코드리뷰도 물론 하고 있습니다.



저희 팀의 특별한 개발 문화를 소개합니다.

MARKET  
*Kurly*

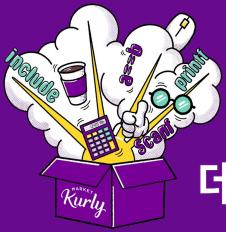
Feature → QA, stage



stage → master



must



세번째 이야기 – 최준영 님

## 데일리 스크럼, 장애 회고 문화도 있답니다.



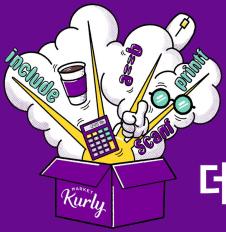
저희 팀의 특별한 개발 문화를 소개합니다.



데일리 스크럼



장애 회고



세번째 이야기 – 최준영 님

## 데일리 스크럼, 장애 회고 문화도 있답니다.



실제 팀원들이 남긴 회고록 - 김포 Picking 서비스 도입을 하며 느낀 점 – 을 공개합니다.



### 오대호 님

회사의 빠른 성장 속도에 따라 다양한 기술적 이슈들을 마주하였고,  
이를 해결해 나가는 과정에서 다양한 경험과 성장을 이룰 수 있었다.  
최고의 복지는 좋은 동료라는 말을 실감했다.  
물심양면으로 지원을 아끼지 않는 동료들과 함께 하지 않았다면 지금과 같이 잘 해낼 수 없었을 것이라 생각한다.  
코로나가 종식되면 오프라인에서 더 많은 이야기를 나누고 싶다.

혼자 하는 것보다 여럿이서 서로 소통하며 의견교환을 하는 것이  
프로젝트를 견고하게 만드는 가장 좋은 방법이라는 걸 느꼈다.  
리팩토링을 진행하며 이전에 작성했던 소스를 객관적으로 평가하고 좀 더 좋게 개선하며 얻는 것이 많았다.  
새로운 개발 방식을 적용시키는데 있어 어려움이 많았다.

이후 개발 시에는 어떻게 하면 좀 더 실무에 맞는 방식이 될지 생각하면서 개발을 해야겠다.



### 홍재성 님



세번째 이야기 – 최준영 님

컬리 합류 독려 메시지



저희 팀에 합류하신다면, 아래의 경험을 얻으실 수 있어요!



수평적인 문화, 누구든 개발에 의견을 낼 수 있고 의견을 낸 내용이 실제로 반영될 수 있어요.

앞서 설명 드린 개발 문화, 기술 스택을 실제로 함께 경험해 볼 수 있어요.

빠르게 성장하는 도중에 많은 변화와 시스템 개선을 진행하는 것을 경험할 수 있어요.

이미 고도화 되어 있는 시스템이 아니기에, 많은 기술들을 고민해보고 도입해 볼 수 있어요.

다양한 버전의 프로젝트를 직접 개발해 볼 수 있어요.

'신규 개발'에 정말 진심입니다!



컬리 개발자성장기

## 네번째 이야기의 주인공

저는

“남편 같은 개발자”

김성준 입니다.



> 김성준님 | 딜리버리프로덕트 플랫폼 개발

[지식 공유, 서비스 라이브러리化를 통해  
폭넓 성장 가능한 컬리 개발 문화]

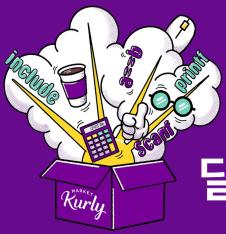
“컬리에는 (누구나, 언제든 원하면) 열린 세미나를 통해  
서로의 지식을 공유할 수 있어요.

그리고 우리 개발자들의 편의를 높이기 위해 서비스들을  
끊임없이 라이브러리화하는 개발 문화도 자리 잡고 있죠.

컬리만의 특별한 개발 문화를 소개합니다.”

MARKET  
*Kurly*

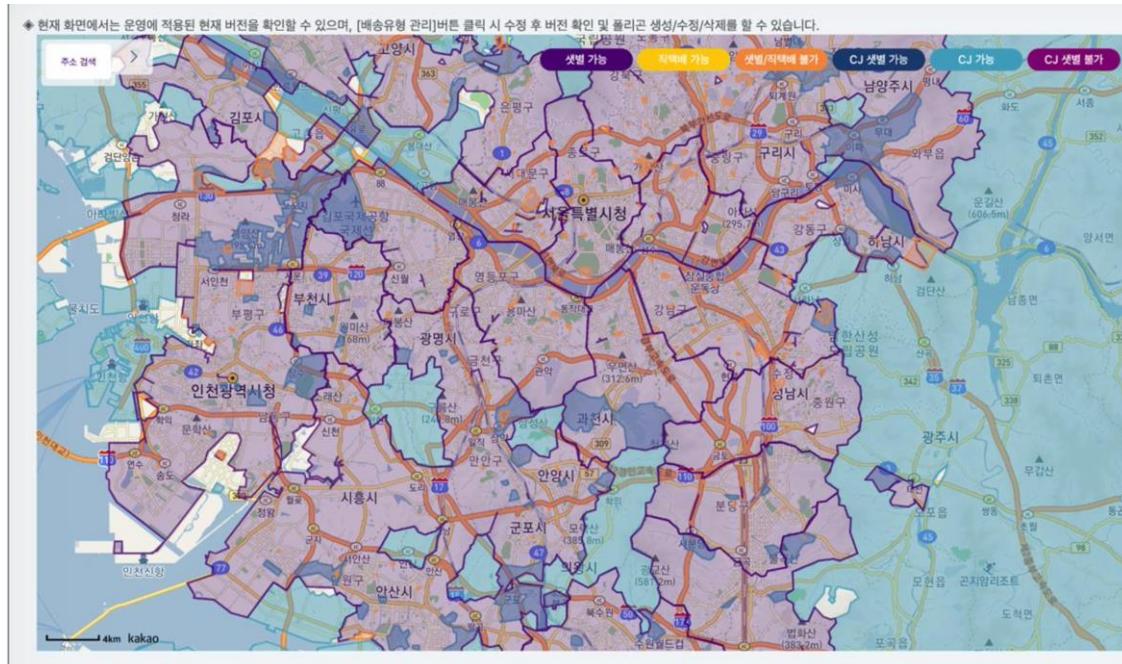




## 네번째 이야기 – 김성준 님 딜리버리 플랫폼 팀 소개



딜리버리 플랫폼 팀은 어떤 팀일까요?



컬리몰의 주문을  
배송 지역, 배송 기사, 비용 등을 고려하여  
최적의 차량 및 배송 경로 탐색!



네번째 이야기 – 김성준 님

## 개발자로서 성장하는 방법

### 개인적인 노력



### 타인과의 교류



이 두가지의 노력이 함께 병행되어야 개발자로서 독보적으로 성장할 수 있죠!



그렇지만, 타인의 노하우와 현장의 경험 습득을 위한 '타인과의 교류'는 혼자서 어찌할 수 없는 것이 현실!



네번째 이야기 – 김성준 님

## 컬리의 개발 지식 공유 문화에 대해



"git 전문가 OOO님! 저희도 전수해주세요!"



"제가 읽은 <오늘의 OOO> 내용이 너무 좋아서, 서평 공유합니다!"

"물론입니다! 지금 바로 모이시죠!"



컬리에서는 최신 트렌드를 주제로 한 세미나 이외에도  
특정 주제에 대한 팀원들의 요청이나 본인이 얻은 개발 지식과 실패담에 대한 다양  
한 주제의 세미나들이 규모를 막론하고 자주 진행됩니다!



도메인 지식 외적으로 정밀 많은 것을 배울 수 있었죠.



네번째 이야기 – 김성준 님

## 개발자들을 위한 개발자로서의 성장



개발자들을 위한 개발자. 무슨 의미인가요?



다른 개발자들을 위해 **라이브러리화, SDK 개발 가이드** 등을 진행하는 개발자입니다.

컬리에서는 자연스러운 생각들



어떻게 하면 다른 팀원들의 손을 조금 덜어 줄 수 있을까?



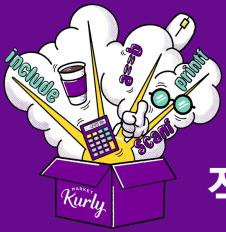
어떻게 하면 소스의 관리 포인트를 최소화 할 수 있을까?

## 공유

라이브러리화

SDK 제작

회고 및 문서 정리



네번째 이야기 – 김성준 님

## 직접 체감한 최고의 사내 라이브러리



실시간 배송 정보 프로젝트 개발 중

배송 정보들을 실시간으로 처리하기 위해  
주문, 배송 업체, 물류 센터 등 여러 곳에서 연동이 필요하네!

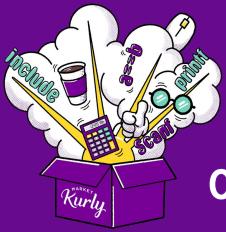
앗! 지금 당장은 결과가 우선적으로 필요한데,  
일일이 구현을 해야 하나?

아니 이미 구현되어 있잖아?!

WOW!



네이버 국회 라이브러리 예시로 이해를 돋기 위한 사진입니다.



네번째 이야기 – 김성준 님

## 이제는 모듈化 된 많은 기능들을 모두 라이브러리화하자!



실시간 배송 정보 프로젝트 개발 중



서비스 간의 연관성이 워낙 깊다 보니  
함께하는 각 팀원분들의 리소스가 과하게 투입되어야 하네!

## 라이브러리화

시간 및 개발 효율 증대를 위해,  
각 서비스들에 필요한 많은 기능들을 모듈화 해보자!

➤ 개인의 개발 시간 축소 & 본인의 개발 부분에 집중 가능



네번째 이야기 – 김성준 님

## 컬리 합류 독려 메시지



어서 빨리 합류하세요!



컬리의 개발 문화를 함께 경험



개인적인 성장과 값진 경험



MARKET  
*Kurly*



컬리 개발자상장기

## 다섯번째 이야기의 주인공

저는

“경계선 의식 없이 일하는 것을 즐기며 문샷 프로젝트에 흥미를 가지고,  
현실 문제를 기술로 풀어본 경험에 성취감을 느끼는 개발자”

최대복 입니다.



> 최대복님 | 데이터 플랫폼 ML 엔지니어

### [물류에서 ML로 최적화 문제를 풀어보기]

“물류데이터를 통해 최적화 문제를 풀어보기 위한 고민의 과정과  
ML을 기반으로 솔루션을 찾기 위한 Deepdive 한 과정을 공유합니다.  
또한 다른 팀원들과 협업하는 방식에 대해서도 함께 소개하고 싶습니다.”

MARKET  
*Kurly*





다섯번째 이야기 - 최대복 님

컬리에서 ML로 풀고자 하는 많은 문제들...

### 마케팅(Marketing)

- 추천
- 검색
- Dynamic Pricing
- ...

### 풀필먼트 (Fulfillment)

- 작업 할당
- 스케줄링
- ...

### 공급망 (Supply Chain)

- 수요 예측
- ...



다섯번째 이야기 – 최대복 님

## 컬리에서 ML로 풀고자 하는 많은 문제들...



각각 문제에 따라서 적합한 모델이 다르죠!

### 마케팅(Marketing)

- 추천
  - : Field-aware Factorization machine, Graph neural network, user embedding
- 검색
  - : Word2Vector, NLP
- Dynamic Pricing
  - : reinforcement learning

### 풀필먼트 (Fulfillment)

- 작업 할당
  - : Genetic algorithm, Meta Heuristic
- 스케줄링
  - : Linear programming

### 공급망 (Supply Chain)

- 수요 예측
  - : prophet, Lstm



다섯번째 이야기 – 최대복 님

## 컬리 풀필먼트 (Fulfillment) 문제에서 경험한 문제들



풀필먼트에서는 어디까지 최적화 할 수 있을까? (Constraint, Boundary condition)

✓ **Boundary Condition**

: 실제로 어떤 부분을 Handling 할 수 있을까? 어디까지 Handling 할 수 있을까?



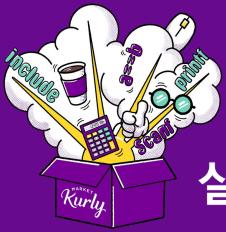
환경이 변할 때 최적화 모델 성능이 유지될까? (Robust & Causality)

- ✓ 물류가 많이 들어오는 시간, 혹은 날짜에 최적 성능을 보장하는 알고리즘이 따라서 달라질 수 있을까?
- ✓ 그렇다면, 외부 환경과 알고리즘 효율의 Causality를 어떻게 알 수 있을까?



풀필먼트 최적화 시스템에서 테스트와 배포는 어떻게 하지? (풀필먼트 시스템에서의 MLOps)

- ✓ 실제 물류 데이터를 받아서 테스트 하는 환경 구축
- ✓ 실제 변경된 최적화 방식이 좋은 쪽으로 변할지, 아니면 더 악화될지 예상할 수 있을까?

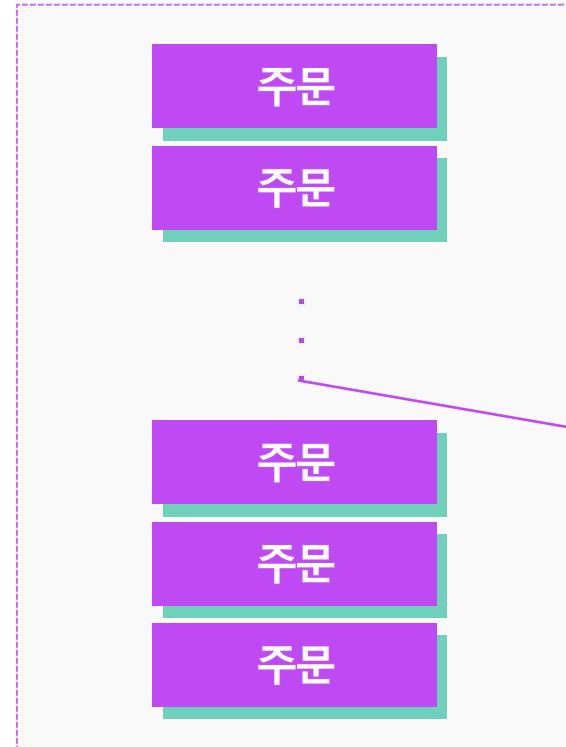


다섯번째 이야기 – 최대복 님

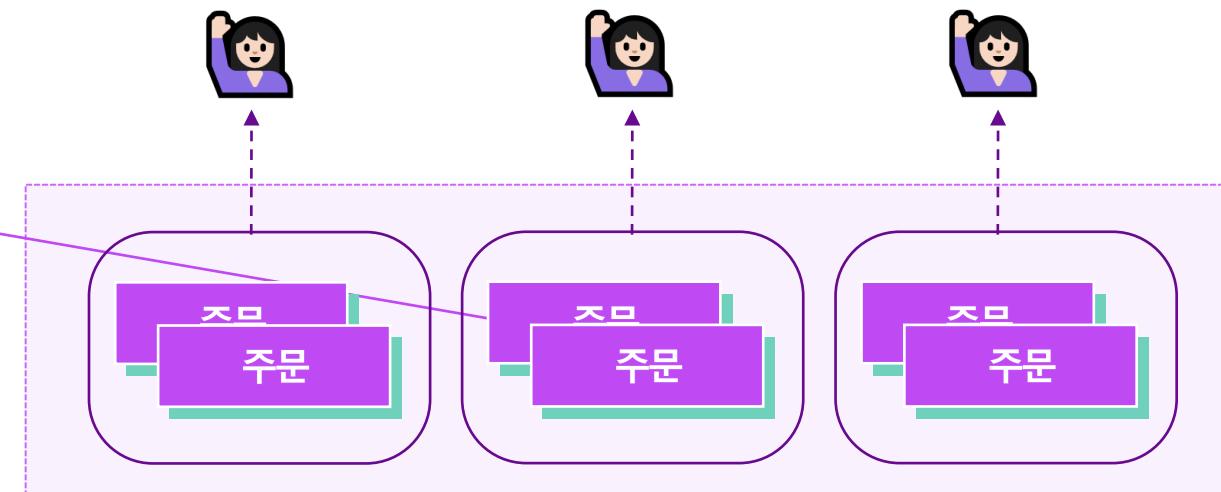
## 실제로 커리 풀필먼트 (Fulfillment) 문제에서 풀어야 할 문제



극도로 단순화 해본 커리 풀필먼트 시스템의 문제는,



➤ 많은 양의 주문이 인입되었을 때,  
적절하게 주문을 묶고(Clustering), 순서를 결정한 뒤(Ordering),  
어떤 작업자에게 할당(Matching) 할지 풀어야 합니다.



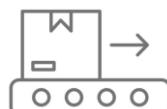
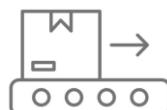
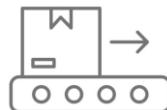


다섯번째 이야기 - 최대복 님

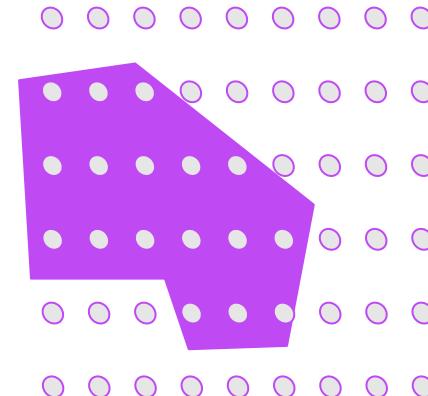
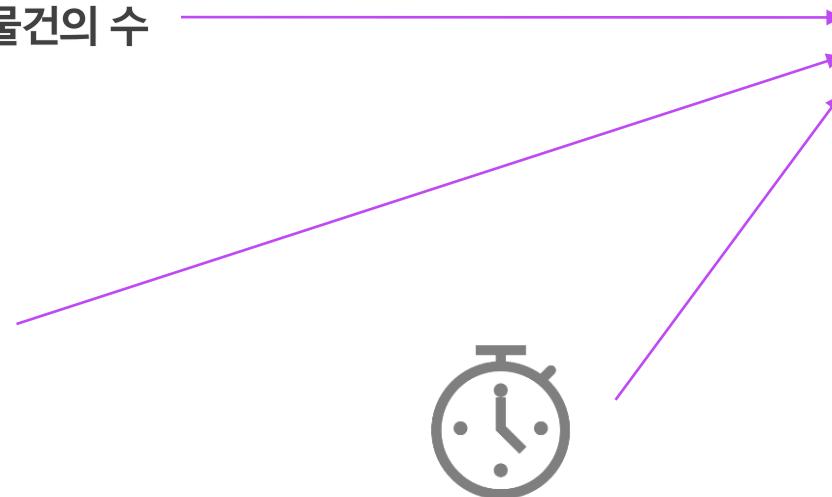
## 풀필먼트(Fulfillment)에서의 Constraint



작업자가 한번에 나를 수 있는 물건의 수



물류 센터에서 병렬 처리할 수 있는 작업의 수



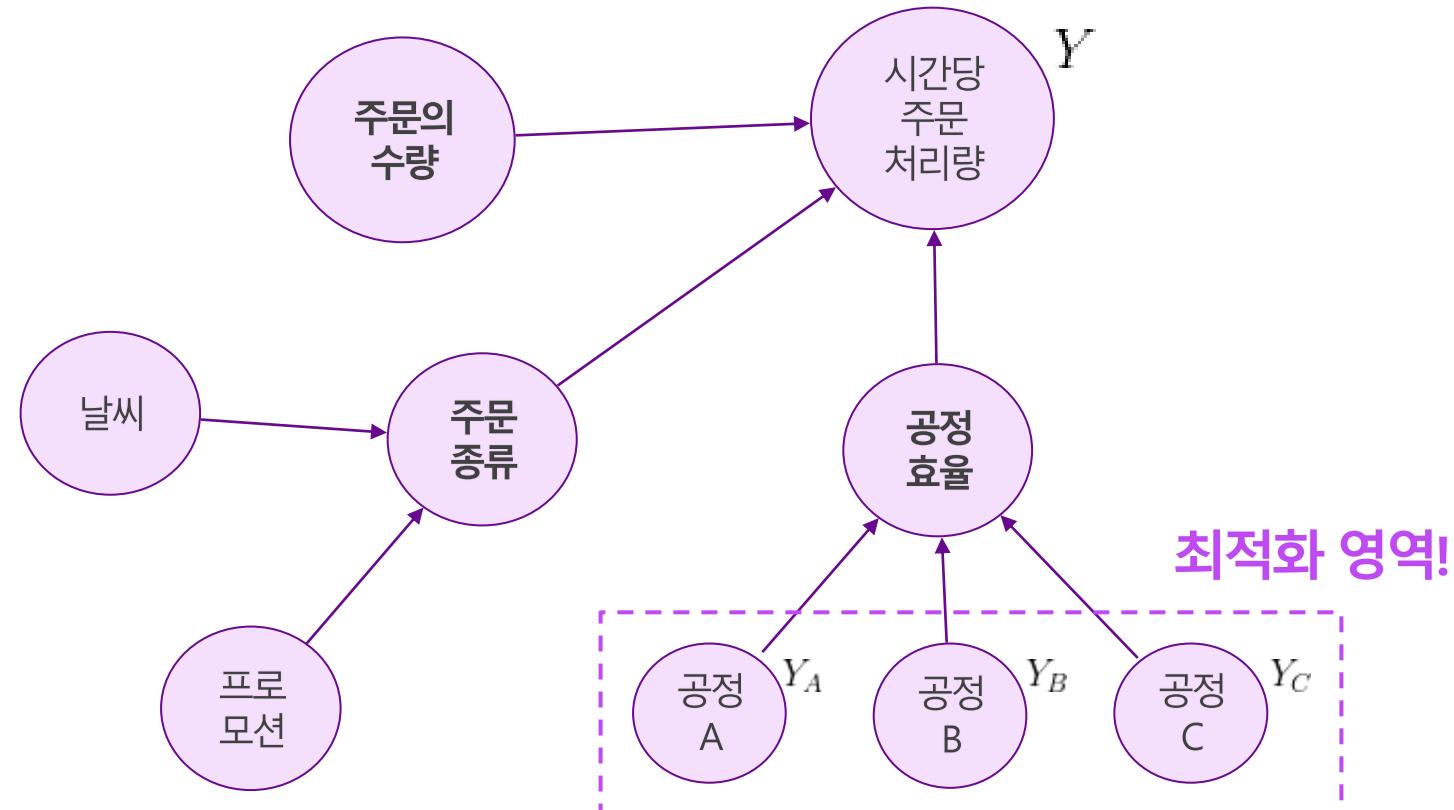
제한(Constraint)이 있는 정수

최적화 요청 처리 되어야 할 최소한의 시간 (with NP-hard)



다섯번째 이야기 – 최대복 님

## 풀필먼트(Fulfillment)에서의 Causality



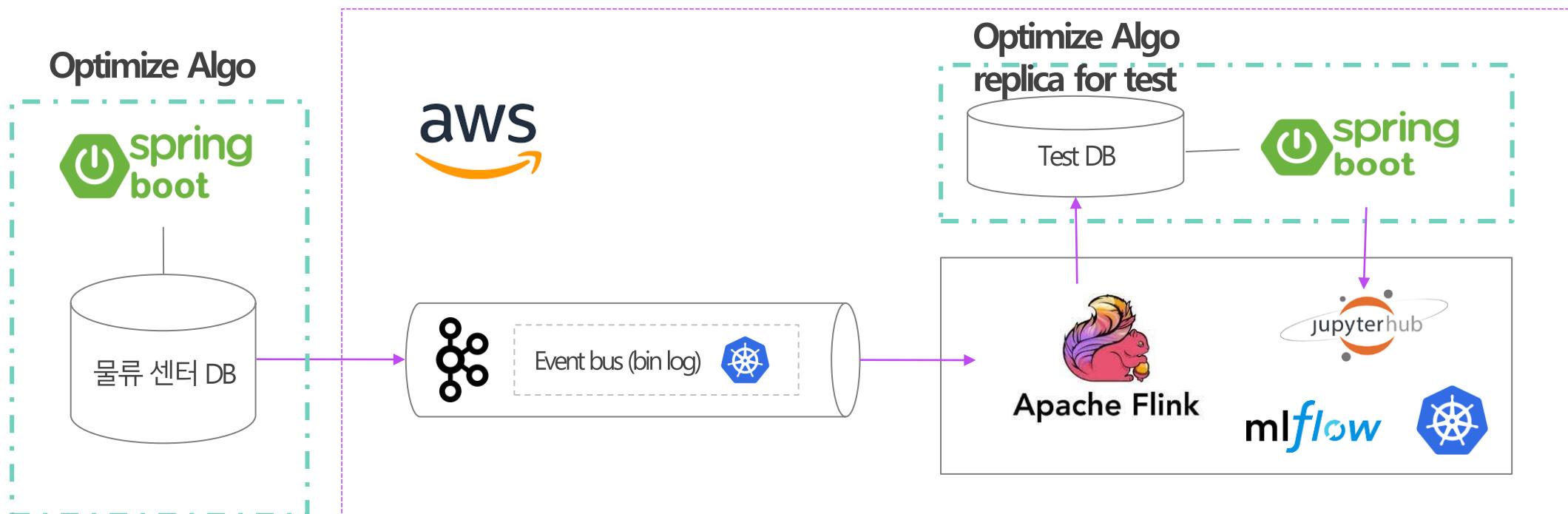


다섯번째 이야기 - 최대복 님

컬리의 테크 Spec을 소개합니다.



컬리의 최적화된 로직은 물류센터 IDC에 구축되어 있습니다!





다섯번째 이야기 - 최대복 님

## 앞으로의 개발 방향

실험 환경은 구축 완료



이제 테스트와 배포를 해보자!



기존 방식보다 향상된 것을 어떻게 알 수 있을까?

(기존 방식보다 향상되지 않았다면 배포X)

- ✓ Causality에 기반한 풀필먼트 최적화 성능 평가하기
- ✓ 컬리의 각 풀필먼트 공정에 적합한 Evaluation metric



풀필먼트(물류) 작업 관리자가 요청하는 시간 내 최적 결과 도출

- ✓ 서비스가 요구하는 response time 맞추기
- ✓ 모든 주문은 개별적인 하나의 작업으로 할당 되어야 함 (Entity integrity)



## 다섯번째 이야기 – 최대복 님

### 컬리 합류 독려 메시지



컬리에 합류하신다면, 아래의 경험을 얻으실 수 있어요!

컬리의 뛰어난 엔지니어들과 함께 일할 수 있어요.

경계선 의식 없이  
공통의 문제를 갖고  
데이터 플랫폼 팀원  
들이 함께 문제에  
참여해요.

고도화 된 풀필먼트  
시스템 위에서 최적  
화를 적용  
해볼 수 있는 기회  
예요.

End to End 관점에  
서 최적화를 고민하  
고 적용해 볼 수 있  
는 환경이예요.

그 외에도 다양한  
컬리가 풀고 있는  
문제들을 해결할  
수 있어요.

- 데이터로 풀필먼트 시스템의 효율성을 측정할 수 있는 경험
- 새로운 알고리즘을 실험해보고 이를 바탕으로 새로운 실험을 설계하는 경험

- 컬리 앱의 주문 첫 시작부터 배송까지 존재하는 수많은 최적화 문제를 고민

- 아이템 추천 (Item Recommendation)
- 수요 예측 (Demand Forecasting)

# MARKET KURLY DEVELOPER MEET UP

FUN 하지만 뻔하지 않은, 컬리의 개발자 성장기

