



APACHE THRIFT

Another way to build API servers

발표자 소개



“처음뵙겠습니다”

배영섭 커머스랩장

Carnegie Mellon University 를 졸업하고
미국 오라클에서 2년 반 동안 근무하다
메쉬코리아라는 스타트업에 매력을 느껴
4년 전부터 함께 일하고 있습니다.

메쉬코리아 연구소 소개

60 명의 임직원이 열정을 불태우고 있어요!



VROONG LAB

부릉은 메쉬코리아의 배송 기사 시스템입니다.
이륜차, 사륜차, 물류센터를 포함한 물류망에 자동 배차 알고리즘을 탑재하여 배송 시간과 비용의 최적화를 이뤄냅니다



PRIME LAB

메쉬프라임은 전국 실시간 배송과 당일 배송을 가능케하는 물류 서비스입니다.
메쉬프라임을 통하면 개인도, 대형 기업도 뭐든지 오늘안에 배송이 가능해집니다



COMMERCE LAB

부탁해!는 배달 안되던 편의점이나 유명 맛집의 음식을 배달하는 서비스입니다.
실시간 추적이 가능한 자체 기사망을 통해 빠르고 편리한 배달이 가능합니다.



SOLUTION LAB

NP-HARD 문제 중 하나인 VRP 를 효과적으로 푸는 TMS 엔진을 개발하고 있습니다.
물류 시스템에 데이터 마이닝 알고리즘들을 도입하고 있습니다.

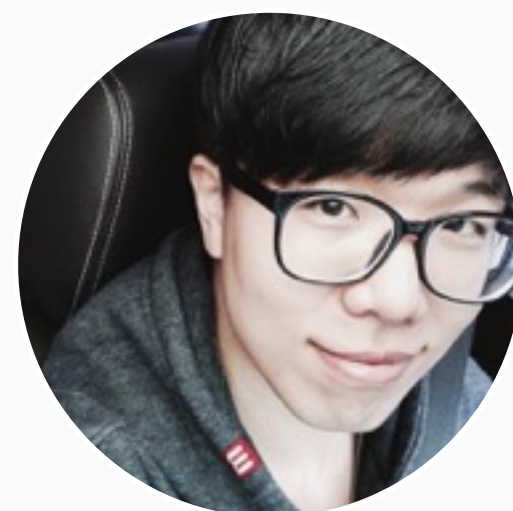
능력자들



이종선
서버개발자



이채은
프론트엔드 개발자



이준
UX 디자이너



김형설
서버개발자



안정수
서버개발자



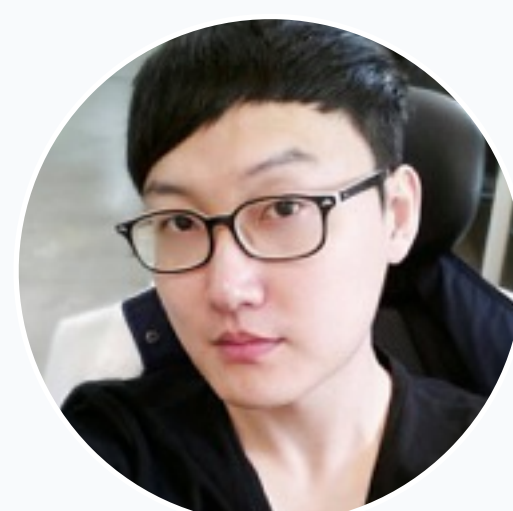
문성렬
UI 디자이너



이지훈
앱 개발자



이미혜
앱 개발자



김대열
WPF 개발자



이민정
그래픽 디자이너

APACHE THRIFT

Another way to build API servers

팀 크기별 플랫폼 간 개발 협업 모습

살펴봅시다!

1인 개발 - API 문서화 따위 필요 없다!



FULL STACK DEV

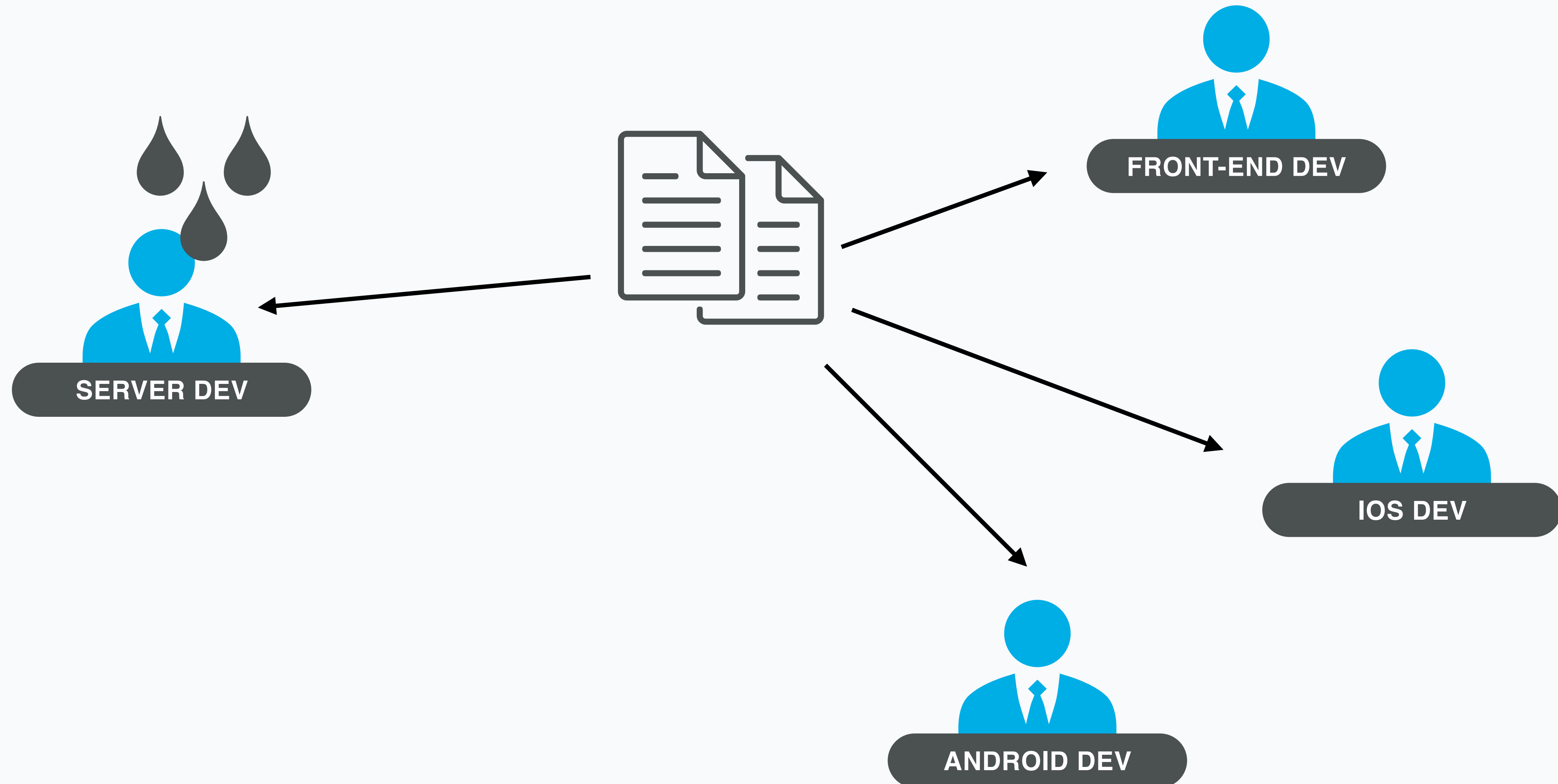
2인 개발 - 대화로 잘 풀어간다



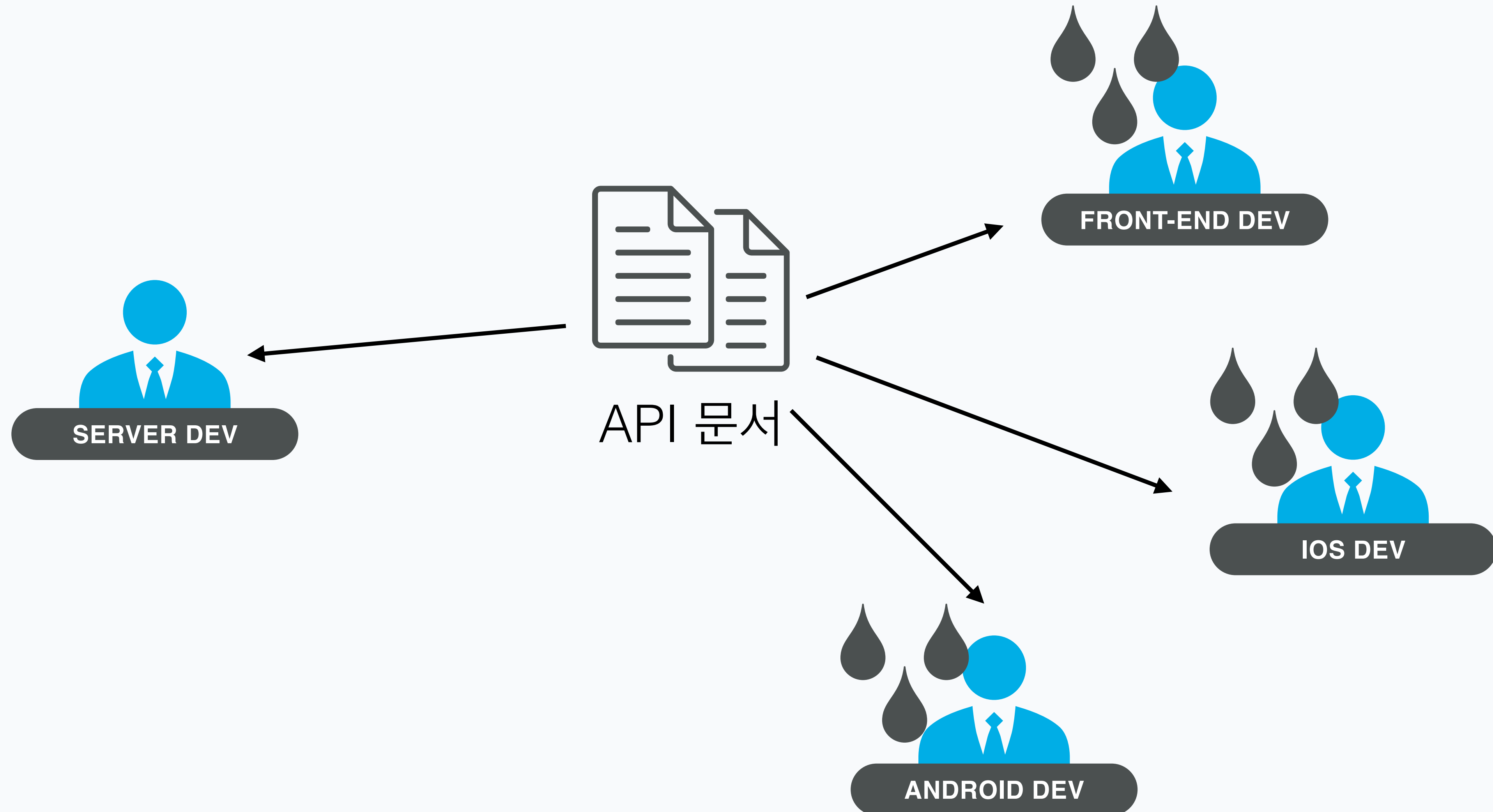
더 많은 클라이언트 개발자 - 서버개발자만 정답을 알고있다



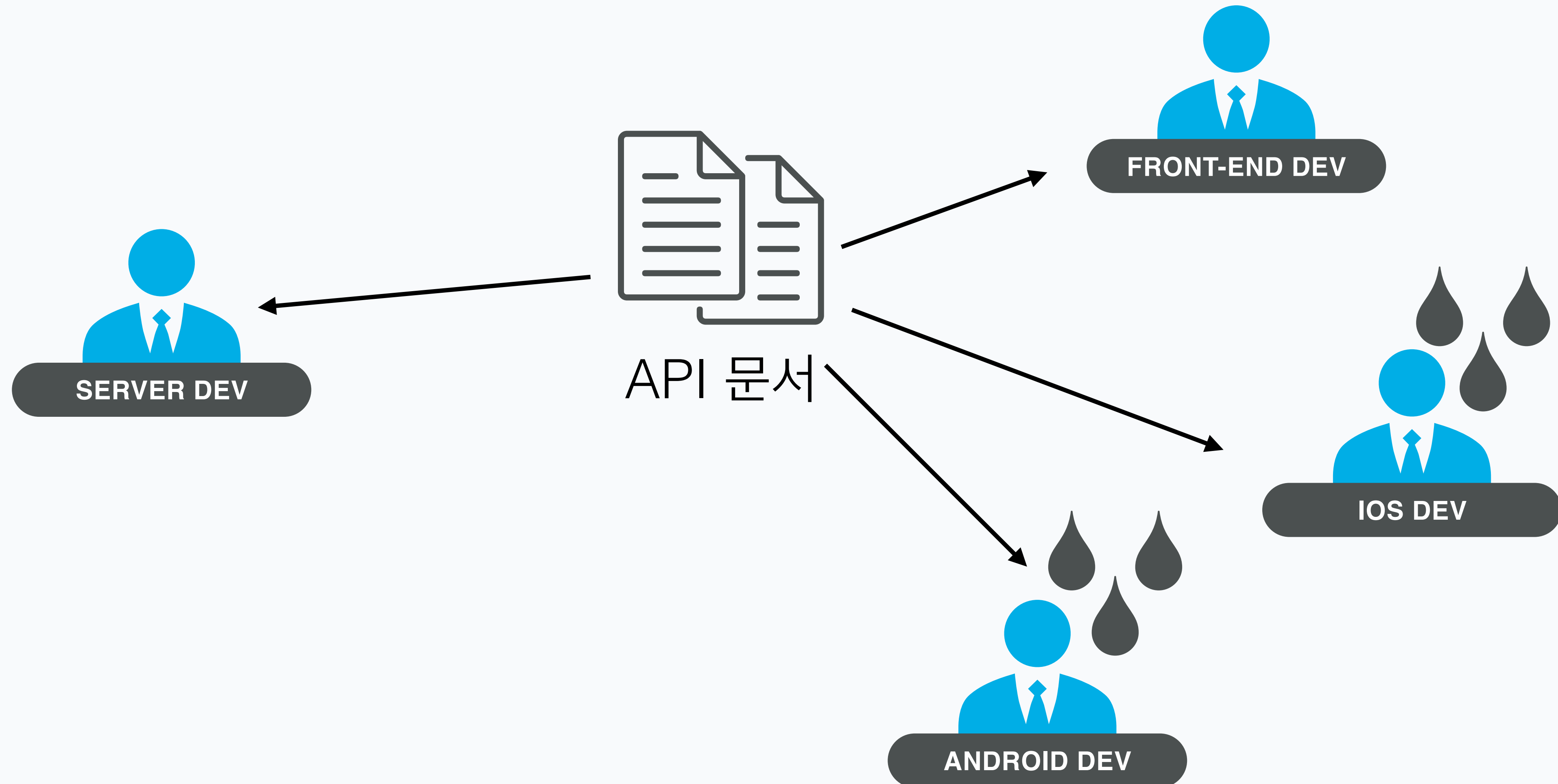
서버개발자가 API 문서를 만들기 시작, 귀찮지만 해야함



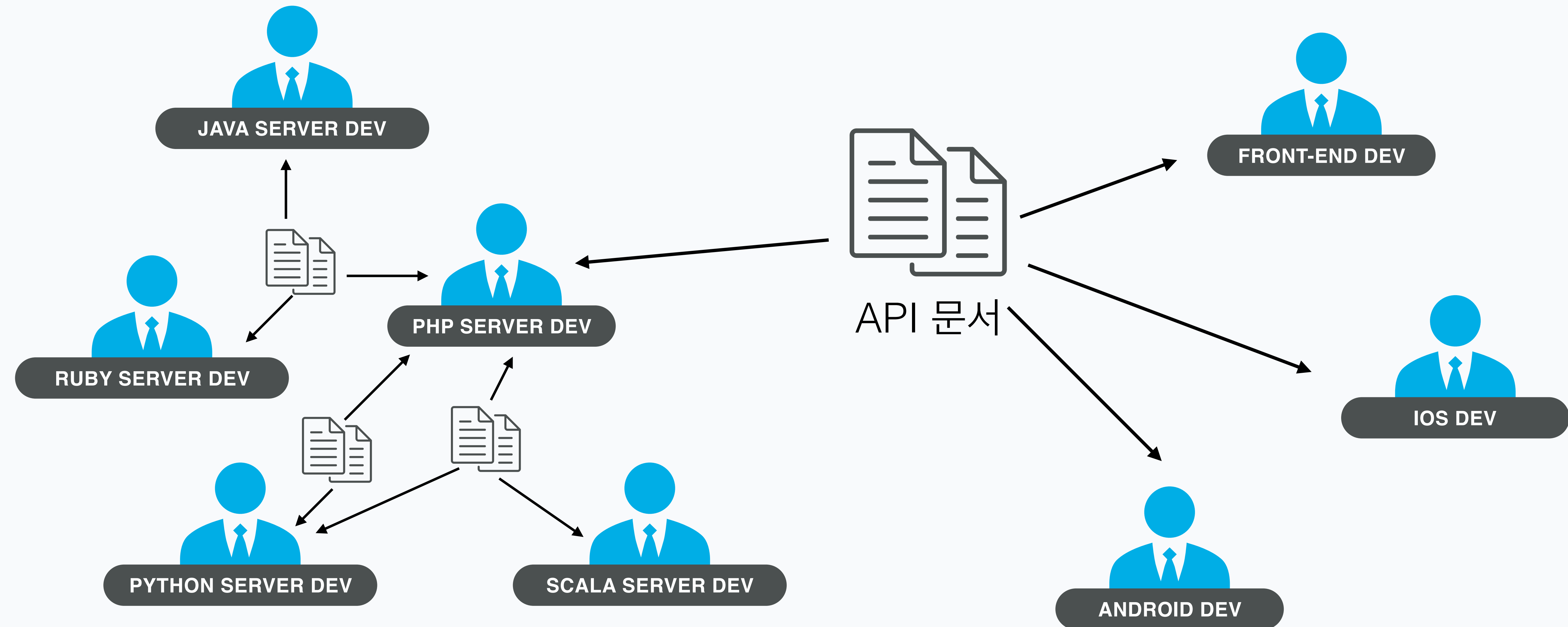
DTO 도 만들고, JSON PARSING 해서 DTO 도 생성하고 ...



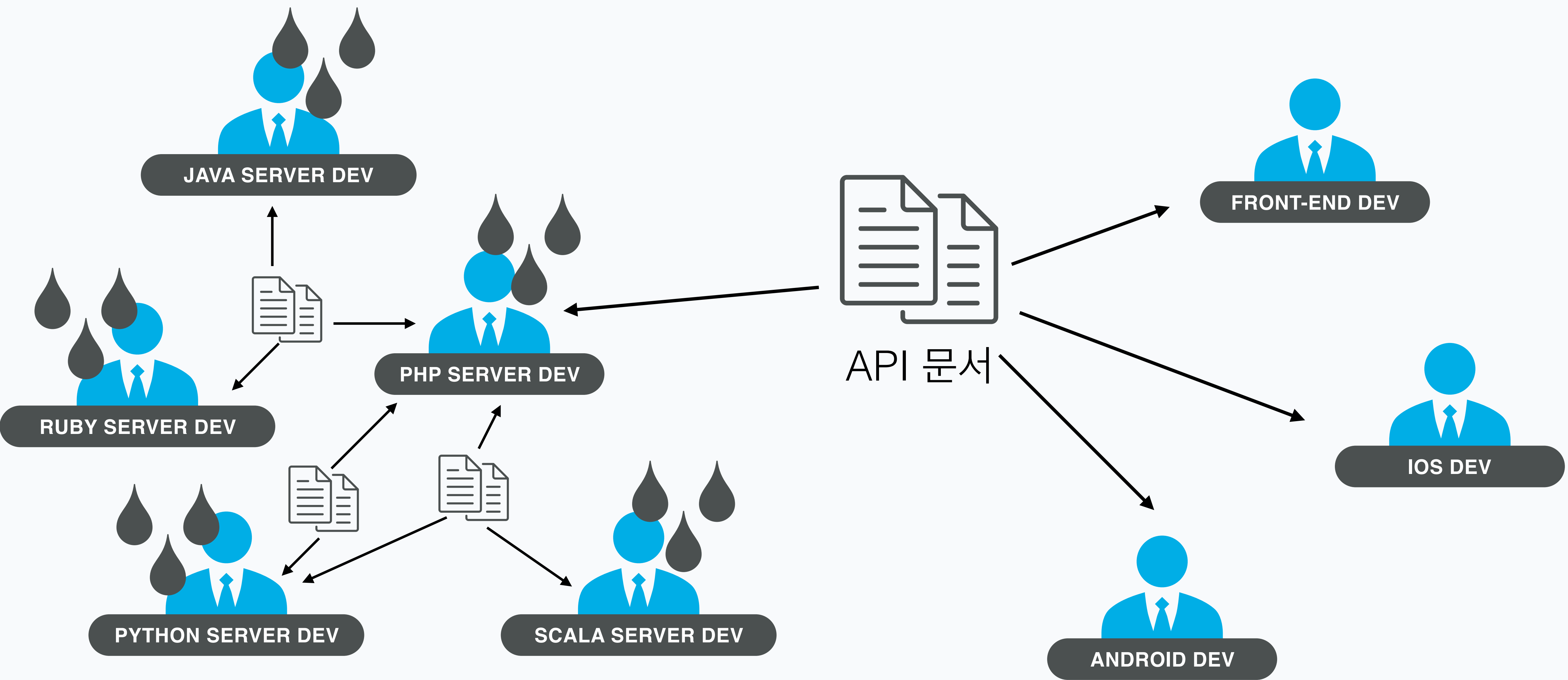
API BREAK CHANGES?! 클라이언트 강제 업데이트



새로운 서버 추가 - 결제 모듈 서버, 추천 서버, 통계 서버, ...



귀찮은 DTO 를 만들고, 귀찮은 DTO 를 만들고, 귀찮은 DTO 를 만들고...



APACHE THRIFT

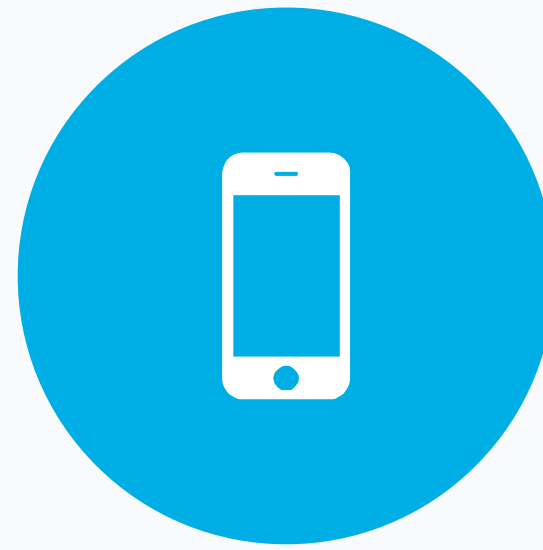
software framework for scalable cross-language services development

FACEBOOK 에서 개발한 THRIFT 특징들



DTO 자동 생성

다양한 언어로 DTO 를 생성해줍니다



버전 관리 용이

Backward Compatibility 를 지원합니다



크기가 작은 메세지

기존 JSON format 보다 작은 크기로
메시지를 주고 받습니다



공통된 IDL

IDL 한번만 배우면 여러 플랫폼을 위해
사용 가능합니다

지원되는 언어 - 매우 많음

PHP

RUBY

ERLANG

C#

PYTHON

JAVA

PERL

COCOA

JAVASCRIPT

C++

HASKELL

DELPHI

SCALA

SMALLTALK

OCAML

쉬운 IDL (INTERFACE DESCRIPTION LANGUAGE)

```
namespace php Mesh.Delivery
namespace js Mesh.Delivery
namespace cocoa MD
namespace java com.mesh.delivery
```

```
/** Timestamp in epoch */
typedef i64 Timestamp
```

```
typedef string AccessToken
```

```
/**
 * 배송 상태
 */
enum DeliveryStatus {
    /** 주문 접수됨 */
    SUBMITTED = 1,
    /** 배송기사가 배정됨 */
    DELIVERING = 2,
    /** 물품을 전달함 */
    COMPLETED = 10,
    /** 배송 취소됨 */
    CANCELLED = 11
}
```

```
enum Gender {
    FEMALE = 1,
    MALE = 2
}
```

```
struct Courier {
    /** 아이디 */
    1: required i32 id,
    /** 이름 */
    2: required string name,
    /** 전화번호 */
    3: required string phone,
    4: optional Gender gender
}
```

```
struct LatLng {
    /** 위도 */
    1: required double lat,
    /** 경도 */
    2: required double lng
}
```

```
struct Delivery {
    /** 주문 번호 */
    1: required string id,
    /** 출발지 */
    2: required LatLng origin,
    /** 도착지 */
    3: required LatLng destination,
    /** 주문 상태 */
    4: required DeliveryStatus status,
    /** 주문 수행 배송기사 */
    5: optional Courier courier,
    /** 접수 시각 */
    6: required Timestamp createdAt
}
```

```
/**
 * 오류 코드
 */
enum ErrorCode {
    /** No information available about the error */
    UNKNOWN = 1,

    /** The format of the request data was incorrect */
    BAD_DATA_FORMAT = 2,

    /** A required parameter/field was absent */
    DATA_REQUIRED = 3,

    /** Access token is invalid */
    INVALID_AUTH = 4
}

exception ServiceException {
    1: optional ErrorCode code,

    2: optional string parameter,

    3: optional string message
}
```

쉬운 IDL (INTERFACE DESCRIPTION LANGUAGE)

```
service DeliveryService {
    Delivery submitDelivery(1: AccessToken accessToken, 2: Delivery delivery)
        throws (1: ServiceException e)

    Delivery getDelivery(1: AccessToken accessToken, 2: i32 deliveryId)
        throws (1: ServiceException e)

    list<Delivery> listDeliveries(1: AccessToken accessToken, 2: i32 offset, 3: i32 limit)
        throws (1: ServiceException e)

    Delivery assignCourier(1: AccessToken accessToken, 2: i32 deliveryId, 3: i32 courierId)
        throws (1: ServiceException e)

    Delivery cancelDelivery(1: AccessToken accessToken, 2: i32 deliveryId)
        throws (1: ServiceException e)

    Delivery completeDelivery(1: AccessToken accessToken, 2: i32 deliveryId)
        throws (1: ServiceException e)
}
```


LARAVEL 에서 THRIFT 연동하기

DeliveryService.php

```
<?php namespace App\Services;

use Mesh\Delivery\Courier;
use Mesh\Delivery\Delivery;
use Mesh\Delivery\DeliveryServiceIf;
use Mesh\Delivery\DeliveryStatus;
use Mesh\Delivery\Gender;
use Mesh\Delivery\LatLng;
use Mesh\Delivery\ServiceException;

class DeliveryService implements DeliveryServiceIf
{
    /**
     * @param string $accessToken
     * @param Delivery $delivery
     * @return Delivery
     *
     * @throws ServiceException
     */
    public function submitDelivery($accessToken, Delivery $delivery)
    {
        return $this->getDeliveryStub();
    }

    /**
     * @param string $accessToken
     * @param int $deliveryId
     * @return Delivery
     *
     * @throws ServiceException
     */
    public function getDelivery($accessToken, $deliveryId)
    {
    }
```

route.php

```
<?php

use App\Services\DeliveryService;
use Mesh\Delivery\DeliveryServiceProcessor;
use Mesh\Thrift\ThriftResponse;
use Illuminate\Http\Request;

Route::post('delivery/{protocol}', function (Request $request, $protocol) {
    $service = new DeliveryService();
    $processor = new DeliveryServiceProcessor($service);
    return ThriftResponse::make($request, $processor, $protocol);
});
```

생성된 문서

Thrift module: Delivery

Module	Services	Data types	Constants
Delivery	DeliveryService <ul style="list-style-type: none"><code>assignCourier</code><code>cancelDelivery</code><code>completeDelivery</code><code>getDelivery</code><code>listDeliveries</code><code>submitDelivery</code>	<code>AccessToken</code> <code>Courier</code> <code>Delivery</code> <code>DeliveryStatus</code> <code>ErrorCode</code> <code>Gender</code> <code>LatLng</code> <code>ServiceException</code> <code>Timestamp</code>	

Services

Service: DeliveryService

Function: DeliveryService.submitDelivery
<pre>Delivery submitDelivery(AccessToken accessToken, Delivery delivery) throws ServiceException</pre>
Function: DeliveryService.getDelivery
<pre>Delivery getDelivery(AccessToken accessToken, i32 deliveryId) throws ServiceException</pre>
Function: DeliveryService.listDeliveries
<pre>list<Delivery> listDeliveries(AccessToken accessToken, i32 offset, i32 limit) throws ServiceException</pre>

Data structures

Struct: Courier

Key	Field	Type	Description	Requiredness	Default value
1	id	i32	아이디	required	
2	name	string	이름	required	
3	phone	string	전화번호	required	
4	gender	Gender		optional	

배송기사

Struct: LatLng

Key	Field	Type	Description	Requiredness	Default value
1	lat	double	위도	required	
2	lng	double	경도	required	

위치 정보

Enumeration: ErrorCode

오류 코드

UNKNOWN	1	No information available about the error
BAD_DATA_FORMAT	2	The format of the request data was incorrect
DATA_REQUIRED	3	A required parameter/field was absent
INVALID_AUTH	4	Access token is invalid

Enumerations

Enumeration: DeliveryStatus

배송 상태

SUBMITTED	1	주문 접수됨
DELIVERING	2	배송기사가 배정됨
COMPLETED	10	물품을 전달함
CANCELLED	11	배송 취소됨

감사합니다!

Contact us:

 서울특별시 강남구 삼성동 162-15 익성빌딩 5층

 010-2826-7262

 youngsub.bae@meshkorea.net