



부정사용거래를 줄이기 위한 통합관리 서비스

하나금융티아이 채용연계형 교육생
심민정

목차

01

SafetyOne 개요

프로젝트 배경

기존 시스템과의 차이점

SafetyOne 목표

02

SafetyOne 서비스 소개

사용자 서비스

안심 서비스 Rule 설정

결제 흐름도

관리자 서비스

03

SafetyOne 구성도

시스템 아키텍처

ERD

04

사용기술

대용량 데이터생성

성능개선

적용된 특화기술

05

시연

시연 시나리오

시연 영상

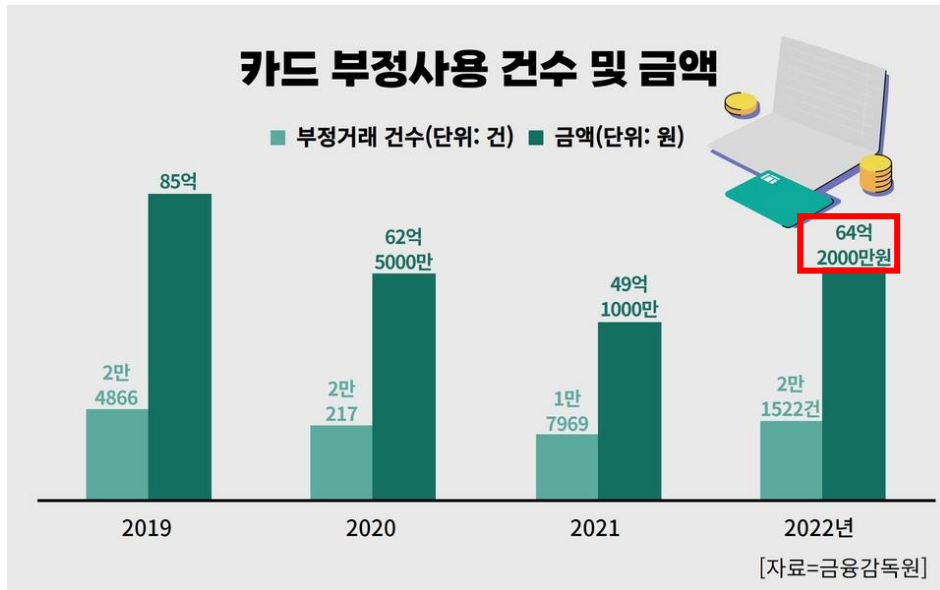
06

보완사항 및 느낀 점

01. SafetyOne 개요

01-1. 프로젝트 배경

신용카드 부정사용 5년간 11.6만건...“관리·감독 강화 필요”



- 2019년부터 최근 4년간 분실·도난, 명의도용 등으로 인한 카드의 부정 사용은 총 8만 4574건이며 **피해액**은 260억 원에 달하는 것으로 나타났다.
- 2018년~2022년까지 5년간 발생한 신용카드 부정사용 내역을 항목별로 살펴보면 **도난·분실이 9만7358건 (88.7%)으로 가장 많았으며** 다음으로 위·변조 9606건 (8.8%), 정보도용 1844건(1.7%) 등으로 이어진다.

=> 이는 곧 **부정거래 금액에 대한 보상**으로 **카드사의 손실**로 이어지고 있다.

부정거래가 일어날 경우의 수를 줄여 **카드사의 손실을 줄이자!**

01-2. 기존 시스템과의 차이점

이상거래탐지시스템(FDS)

사후 관리에 중점

결제 승인 이후 발생한 부정거래 데이터를
분석하여 이상거래를 탐지



사전 + 사후 관리

결제 승인 이전 사용자가 설정한
Rule과 비교하여 결제를 사전에 차단



피해 발생 전에 거래 차단

01-3. SafetyOne 목표

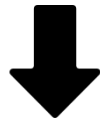
부정사용거래를 줄이기 위한 사전, 사후 통합 관리 서비스

사전



안심서비스

거래를 허용하거나 차단할 **본인만의 Rule**을 **설정**하여 승인요청내역이 Rule에 해당하는 거래는 **사전**에 차단



부정거래의 가능성을 **사전**에 **예방**하며, 사용자에게 **안전한 거래 환경**을 제공

사후



이상소비 알림 서비스

거래 정상승인 **이후** 결제 건이 사용자의 평소거래 **패턴과 다를 경우** 고객에게 **실시간**으로 알림 메시지 전송



부정사용의 **연속적인 발생**을 방지

02. SafetyOne 서비스 소개

02-1. 사용자 서비스



02-2. 안심 서비스 Rule 설정

안심 Rule 설정

Step 1



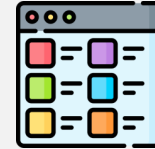
허용 지역 선택

Step 2



차단 시간 선택

Step 3



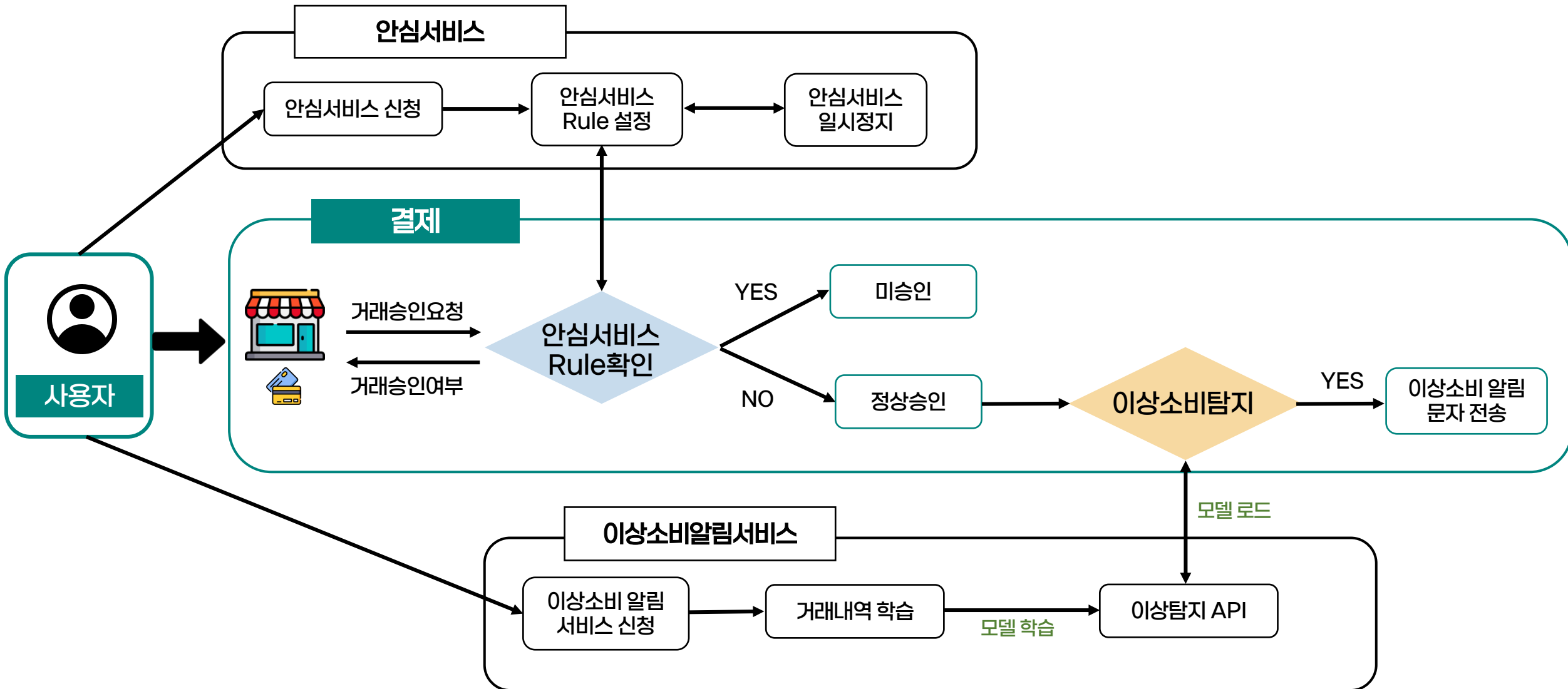
차단 업종 선택

서비스 일시정지

안심 Rule 차단 항목 중 일정기간동안
거래를 허용하도록 서비스 일시정지

02. SafetyOne 서비스 소개

02-3. 결제 흐름도



02. SafetyOne 서비스 소개

02-4. 관리자 서비스



군집분석

- ✓ 하나카드 이용자 소비데이터 확인
- ✓ 군집 별 특징 확인



이메일 전송

- ✓ 군집 별 안심서비스 추천 메일 전송



이상소비알림서비스 관리

- ✓ 이상소비내역 확률분포 시각화
- ✓ 서비스 사용자 및 이상소비내역 관리



안심서비스 관리

- ✓ 서비스 사용자 및 미승인 내역 관리



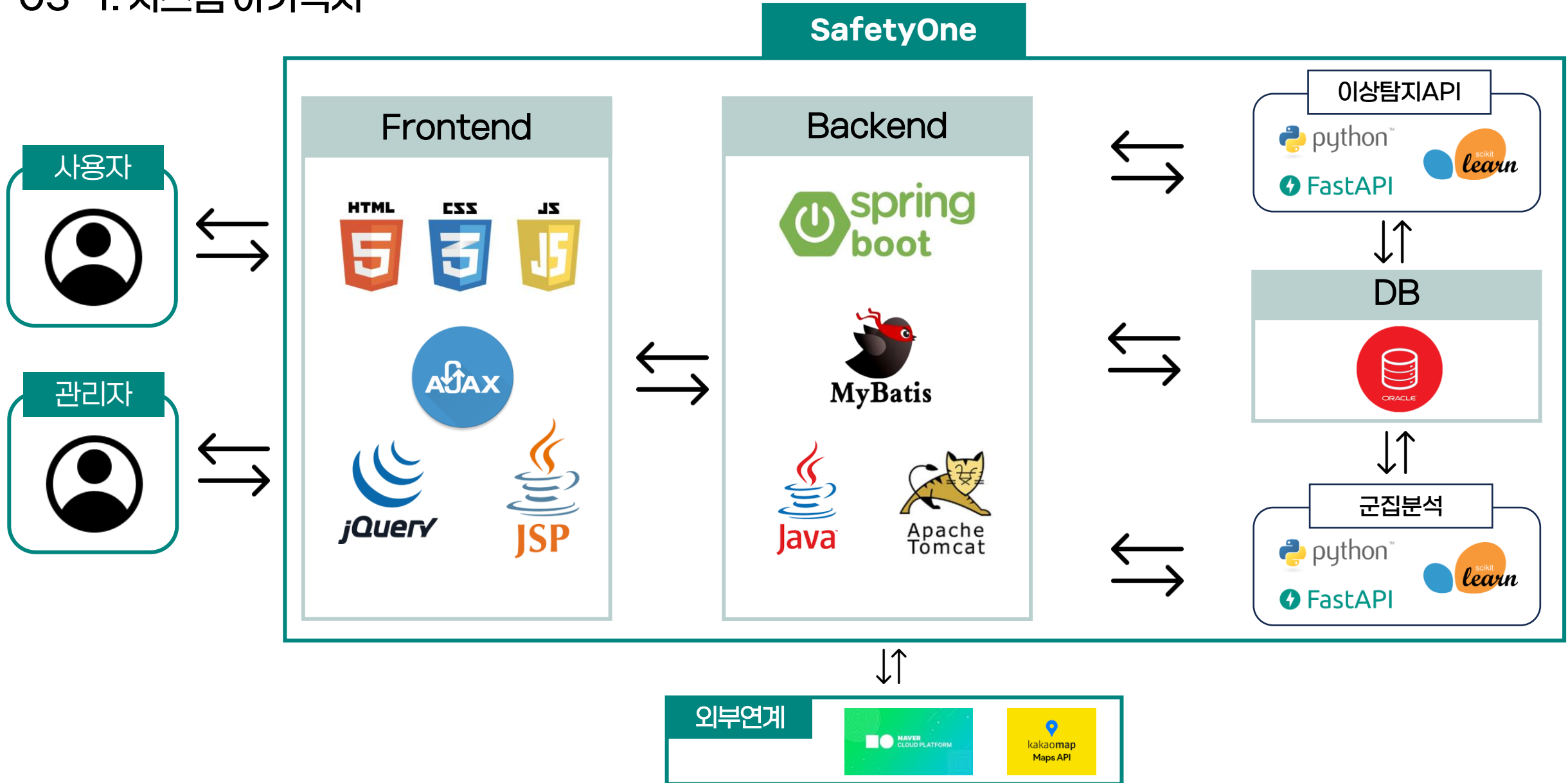
분실/도난 신고 관리



결제 로그 관리

03. SafetyOne 구성도

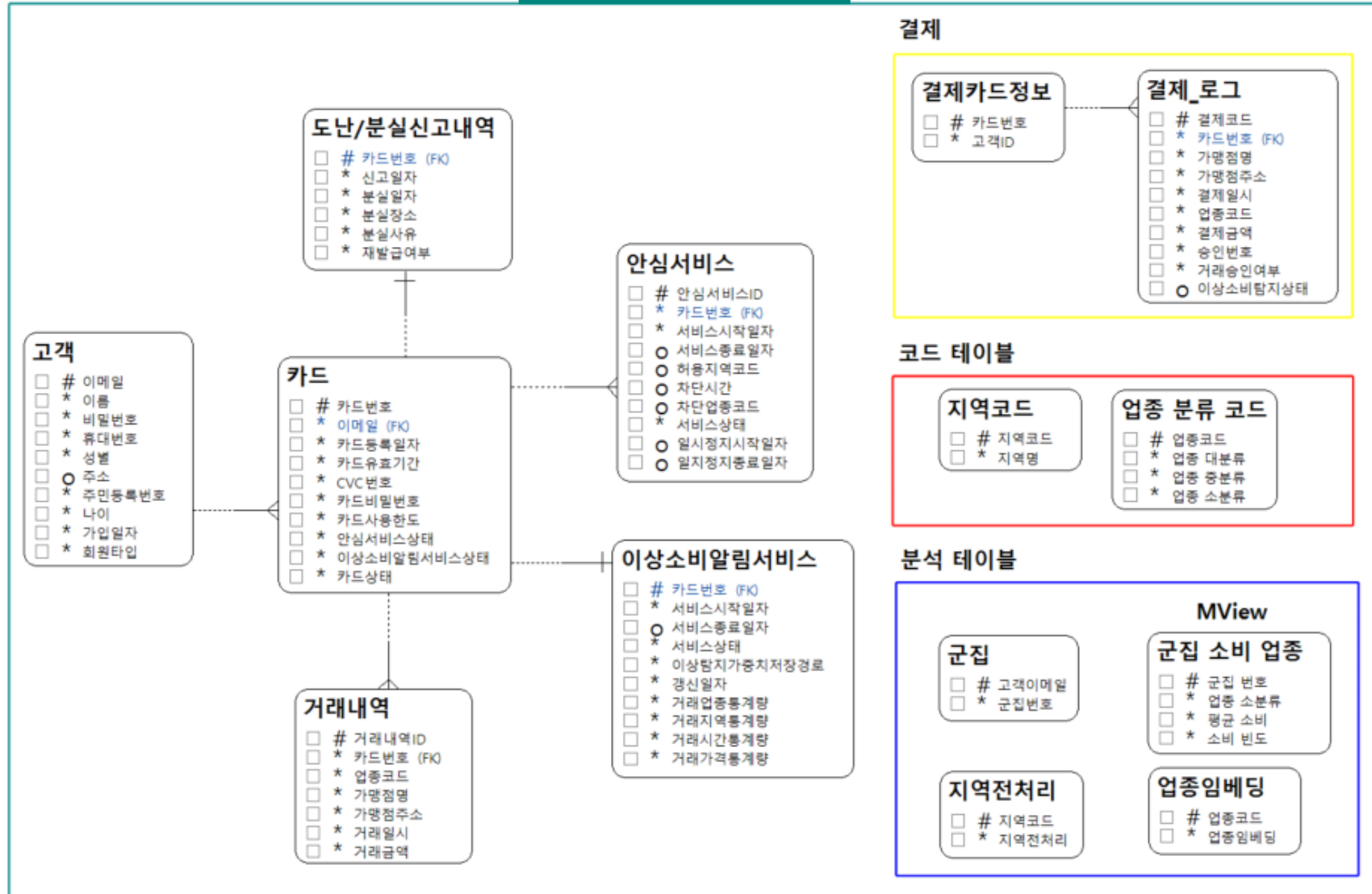
03-1. 시스템 아키텍처



03. SafetyOne 구성도

03-2. ERD

SafetyOne



04. 사용기술

04-1. 대용량 데이터 생성

서울시 빅데이터 캠퍼스에서 제공하는 **신한카드 고객 소비 통계자료** 기반

고객 약 7,000명을 대상으로 2022.10 ~ 2023.10까지 총 500만 건의 가상의 소비 데이터 생성



고객 데이터 생성

20~80세까지

약 7,000명의 고객 데이터 생성



카드 데이터 생성

1인당 약 3개의 카드 소유

약 22,000개의 카드 데이터 생성



소비 데이터 생성

연령, 성별에 따라 소비 가중치 부여,
업종별 소비 시간 고려

약 500만 건 소비 데이터 생성

04. 사용기술

04-2. 성능개선

INDEX를 활용한 쿼리문 시간 단축

스크립트 출력 x 질의 결과 x

SQL | 50개의 행이 인출됨(2,466초)

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		630	92610	20585 (1)	00:00:01
1	SORT ORDER BY		630	92610	20585 (1)	00:00:01
* 2	FILTER					
* 3	HASH JOIN		630	92610	20584 (1)	00:00:01
* 4	TABLE ACCESS FULL	CARDHISTORY	631	58052	20481 (1)	00:00:01
* 5	TABLE ACCESS FULL	CARD	1124	61820	102 (0)	00:00:01

쿼리문 실행 : 2.46초 → 0.17초

스크립트 출력 x 질의 결과 x

SQL | 50개의 행이 인출됨(0.17초)

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		630	92610	1374 (1)	00:00:01
1	SORT ORDER BY		630	92610	1374 (1)	00:00:01
* 2	FILTER					
* 3	HASH JOIN		630	92610	1373 (1)	00:00:01
4	TABLE ACCESS BY INDEX ROWID BATCHED	CARDHISTORY	631	58052	1271 (1)	00:00:01
* 5	INDEX RANGE SCAN	IDX_CARDHISTORY_OPTIMIZED	631		640 (1)	00:00:01
* 6	TABLE ACCESS FULL	CARD	1124	61820	102 (0)	00:00:01

M-VIEW를 활용한 쿼리문 시간 단축

질의 결과 x

SQL | 50개의 행이 인출됨(2,407초)

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	Time
0	SELECT STATEMENT		174	19836		32960 (2)	00:00:02
1	SORT ORDER BY		174	19836		32960 (2)	00:00:02
2	HASH GROUP BY		174	19836		32960 (2)	00:00:02
* 3	HASH JOIN		5716K	621M		32648 (1)	00:00:02
4	VIEW		5	75		10 (10)	00:00:01
5	HASH GROUP BY		5	10		10 (10)	00:00:01
6	TABLE ACCESS FULL	CLUSTERTABLE	7343	14686		9 (0)	00:00:01
* 7	HASH JOIN		5071K	478M		32622 (1)	00:00:02
8	TABLE ACCESS FULL	CLUSTERTABLE	7343	157K		9 (0)	00:00:01
* 9	HASH JOIN		5083K	373M	1144K	32600 (1)	00:00:02
10	TABLE ACCESS FULL	CARD	22485	878K		102 (0)	00:00:01
11	TABLE ACCESS FULL	CARDHISTORY	5155K	181M		20473 (1)	00:00:01

쿼리문 실행 : 2.4초 → 0.026초

질의 결과 x

SQL | 50개의 행이 인출됨(0.026초)

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		245	5880	3 (0)	00:00:01
1	MAT_VIEW ACCESS FULL	CLUSTERSTATIC	245	5880	3 (0)	00:00:01

04. 사용기술

04-3. 적용된 특화 기술



군집분석

STEP1

3개월 거래내역 데이터
전처리

STEP2

K-MEANS
CLUSTERING 학습

STEP3

군집분석



이상탐지API 개발

STEP1

6개월 거래내역 데이터
전처리

STEP2

GMM(Gaussian Mixture
Model) 모델 학습

STEP3

새로운 거래 데이터 입력

STEP4

학습된 분포와 비교하여
이상소비 탐지

request : 새로운 거래 데이터

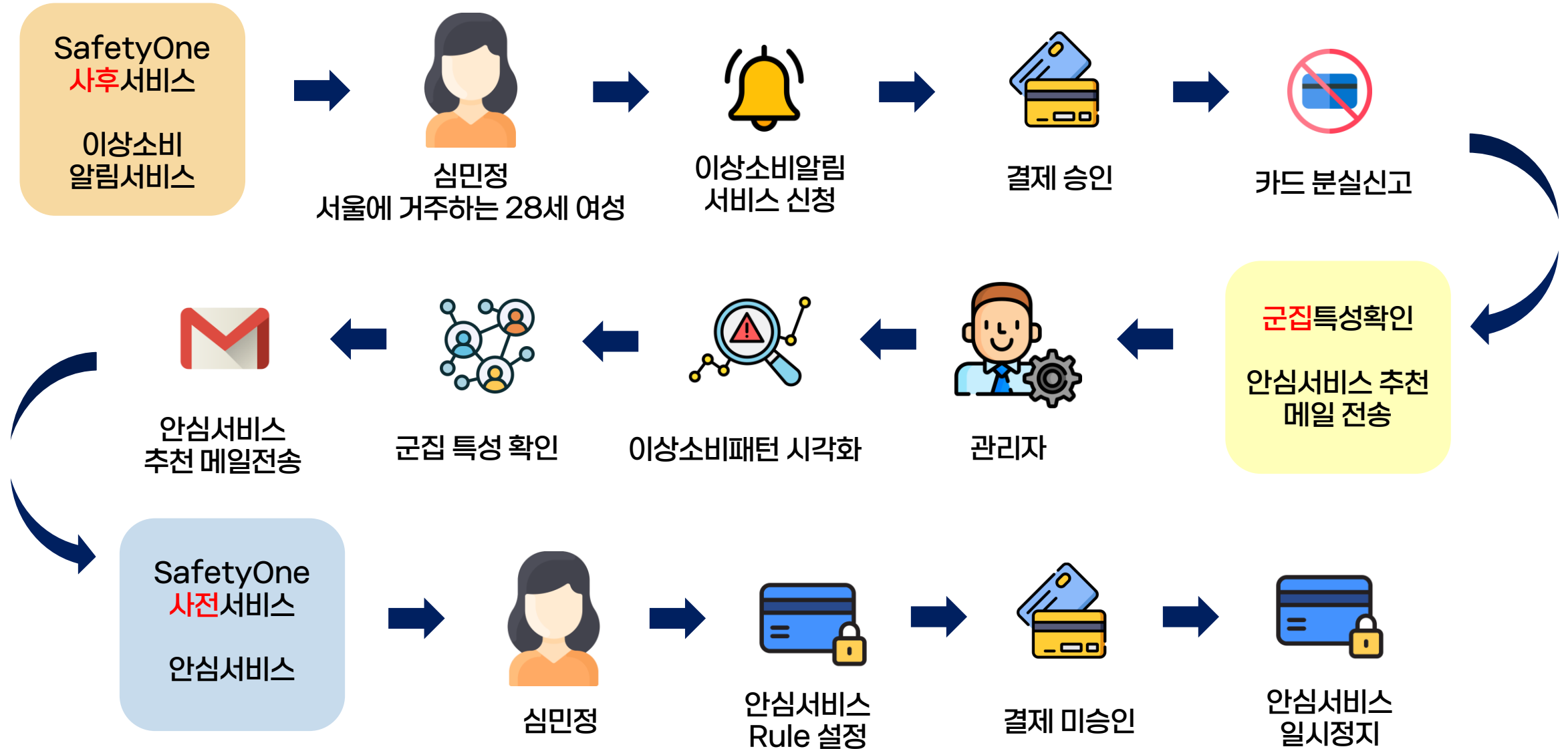
```
{ "wordToVec": {  
  ... 업종  - "categorySmallNumeric": -0.8,  
  ... 시간  - "timeNumeric": -18,  
  ... 지역  - "regionNameNumeric": -163,  
  ... 금액  - "amountNumeric": -500000  
  ... },  
  ... "cardId": "9440-0323-6869-8530"  
}
```

response : 이상소비유무

1 "Y" 1 "N"

05. 시연

05-1. 시연 시나리오



05. 시연

05-2. 시연 영상

보완사항

실시간 다량의 트랜잭션 처리의 아쉬움

→ 다량의 결제 데이터 처리시 과부하 발생 우려

실제 데이터의 사용

→ 실제 데이터를 사용함으로써 정확한 인사이트 도출

느낀 점



하나금융티아이에서 꾸준히 성장하는 개발자

감사합니다.

개발일정

개발 프로세스		9/4	9/11	9/18	9/25	10/2	10/9	10/16	10/20
분석	주제 선정								
	요구사항 분석								
설계	기능 설계								
	ERD								
	아키텍처 설계								
구현	데이터 생성								
	군집분석								
	이상탐지 API 개발								
	회원 기능 구현								
	결제 기능 구현								
	관리자 기능 구현								
테스트	기능 테스트 및 성능 개선								
발표	PPT 제작 및 영상 제작								
	발표준비								

04. 사용기술

04-4. 그 외 적용기술

스케줄러

- 안심서비스의 일시정지 기능은 시작일자와 종료 일자에 따라 자동으로 서비스 상태를 변경
- 군집분석 3개월 주기로 시행

Naver SMS

- 이상소비로 탐지될 경우, 사용자에게 알림 메시지 전송
- 안심서비스로 거래가 미승인될 경우, 사용자에게 메시지 전송
- 사용자 인증 시 필요한 인증번호 메시지 발송

카카오 Map API

- 오프라인 결제 구현을 위해 거래할 가맹점 검색

Java Mail Sender

- 군집 별 안심서비스 차단추천 업종을 사용자에게 이메일로 전송

ChartJs

- ChartJs를 통해 시각화와 정보 전달

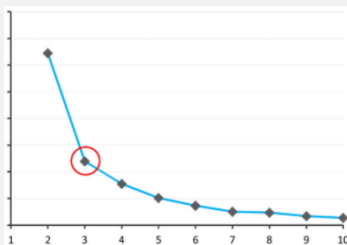
군집분석 프로세스

Step 1



약 50개의 업종
데이터 전처리 및
임베딩

Step 2



Elbow-Method에
따라 데이터를 가장 잘
설명하는 주성분 개수
결정

Step 3



K개의 주성분을
K-means Cluster로
군집의 특성 파악

Step 4



각 군집에 소속된 회원
대상으로 안심서비스
추천