

10장 - 알림시스템설계

알림 시스템은 일반적으로 mobile push notification, sms message, email 세가지로 분류된다.

1. 문제 이해 및 설계 범위 확정

- 종류 : 푸시 알림, SMS, 이메일
- 분류 : 연성 실시간 시스템 - 약간의 지연 무방
- 단말 : ios, android, laptop, desktop
- 알림 주체 : 클라이언트 애플리케이션, 서버측 스케줄링
- 알림을 받지 않도록 opt-out 설정 - 가능
- 하루에 몇 건 보낼 수 있어야하는지?
 - 모바일 푸시 알림 : 천만건
 - SMS : 100만건
 - 이메일 : 500만건

2. 개략적인 설계안 구하기

일반적으로 알림 제공자가 중간 원격 서비스(APNS, FCM, SMS, Email Service) 등을 이용하여

```
flowchart LR
    NP[Notification Publisher] --> NPS
    subgraph NPS
        direction LR
        APNS
        FCM
        SM[SMS Service]
        ES[Email Service]
    end
    NPS --> Device
```

연락처 정보 수집 절차 : 단말 → LB → API Servers → Database 로 관리한다.

- 테이블 구성 : user [phone number, email] < device [id, device token]

또한 단일 서비스로 구현시 문제점

- SPOF : 단일 실패 지점 - 전체 서비스 장애로 이어질 가능성이 크다.
- 규모 확장성 : 한대 서비스로 푸시 알림에 관계된 모든 것을 처리함 - 데이터베이스, 캐시 등 주요 컴포넌트를 개별적으로 늘릴 수 없음
- 병목 : 모든 것을 한 서비스로 처리하면 시스템에 과부하 발생 가능

개략적인 설계안

- 데이터베이스, 캐시를 주 서버에서 분리
- 수평적 규모 확장이 이뤄질 수 있도록 구성
- 메세지 큐를 이용해 시스템 컴포넌트 사이의 강한 결합 제거

flowchart LR

OS[1...N Service] --> NS[Notification Service]

NS --> DB[(Database)]

NS --> Cache(Cache)

subgraph Workers

direction LR

QSMS[{Queue SMS}] --> SmsWorker

QEMAIL[{Queue EMail}] --> EmailWorker

QFCM[{Queue FCM}] --> FcmWorker

QASPN[{Queue ASPN}] --> ASPNWorker

end

NS ...-> QSMS

NS ...-> QEMAIL

NS ...-> QFCM

NS ...-> QASPN

SmsWorker ...-> SMSProvider

EmailWorker ...-> EmailProvider

FcmWorker ...-> FCM

ASPNWorker ...-> ASPN

3. 상세 설계

1. reliability

- 데이터 손실 방지 : 작업서버에서 재시도 매커니즘을 구현할 수 있도록 전송 로그를 보관하는 데이터베이스를 유지한다.
- 알림 중복 전송 방지 : 이벤트 ID를 검사하여 이전에 작업한 적이 있는 이벤트인지 살피는 방식

2. 추가로 필요한 컴포넌트 및 고려사항 : rate limiting, retry mechanism, security, queue

- a. 알림 템플릿 : 일관된 알림 형식을 유지하기 위해서 적절한 알림 템플릿을 미리 구성해두는 것이 좋다
- b. 알림 설정 : 사용자가 알림을 상세히 조절할 수 있도록 관련된 설정을 제공한다.
- c. 전송률 제한 : 한 사용자에게 전송되는 알림을 제한(조절)한다.
- d. 재시도 방법 : 실패시 재시도 큐에 넣는다
- e. 큐모니터링 : 큐의 메트릭을 체크해서 작업 추이를 확인한다.
- f. 이벤트 추적 : 데이터 분석 서비스를 이용해서 알림 확인률, 클릭율, 실제 앱 사용 이어지는 비율 등과 같은 메트릭을 이해하는 것이 중요하다.