

이마트 AdTech, 개발 스터디

1. 스터디 기본 규칙

- 스터디는 본격적인 개발 업무에 대한 “자연스러운 온보딩” 을 목표로 합니다.
- 해당 과정은 초심자를 위한 것이며, 강제성이 없으므로 본인 의사에 따라 참여하지 않으셔도 됩니다.
- 아래 과정은 개발과 회고를 반복하며 본인 스스로 성장하는 과정이 되었으면 좋겠습니다.
- 타인의 회고 과정을 청취하고 질문/답변의 시간을 공유 하면서 본인의 과정을 되돌아 보는 성찰을 하는 시간이지 결과물을 가지고 부족한 점을 평가하는 과정은 아닙니다.
- 누구든 스터디에 참여 할 수 있으며, 누구든 스터디를 중단 할 수 있습니다.

2. 스터디 가이드

- 스터디 1차 과제
 - 목표 가이드
 - Spring Boot 환경 하에서 실무에 사용할 미들웨어 연동을 합니다
 - 본인의 수준에 맞추어 아래와 같은 시나리오도 추천 드립니다.
 - MVC, WebFlux 각각의 환경을 동시에 혹은 하나만 구현 하셔도 됩니다.
 - MVC, WebFlux 동시에 구현하시고 벤치마킹 과정을 공유 하셔도 됩니다.
 - MVC, WebFlux 를 구성하고 벤치마킹 도구로 각각 아키텍처의 성능을 측정합니다.
 - 어떤 아키텍처든 게시판이라는 큰 주제를 벗어나지 않는 이상 원하시는 별도의 목표를 실천하시고 공유 하셔도 됩니다.
 - 일정 가이드
 - 11월 06일 개발
 - 아래 기능을 MVC 환경하에 MariaDB 를 연동하여 개발한다.
 - 11월 07일 개발
 - 아래 기능을 WebFlux 환경으로 프로젝트를 변경 하고 MongoDB 를 추가 연동하여 개발한다.
 - 11 월 08일 회고
 - 각자의 결과물을 가지고 회고의 시간을 가지도록 한다.
- 스터디 2차 과제

: 1차 이후 목표 및 일정 변동 가능함

 - 목표 가이드
 - 1차 과제의 결과물에서 Cache 를 연동 하도록 구성하고 벤치마킹 도구로 성능을 측정합니다.
- 스터디 3차 과제

: 1차 이후 목표 및 일정 변동 가능함

 - 목표 가이드

- 2차 과제의 결과물에서 Kafka Producer 를 추가하고 Consumer 를 개발해서 외부 API 를 연동합니다.
(이때 외부 API 는 또 다른 Table 에 저장을 하는 API 입니다)

3. 기술 스택 및 기본 API 구성

- 기술 스택
 - DB
 - MariaDB
 - MongoDB
 - Cache
 - Redis
 - Queue
 - Kafka
- 게시판 명세
 - 공통 Response 규격
 - code
 - message
 - data
 - Board API
 - 저장
 - POST
 - /v1/board/item
 - title
 - body
 - modified
 - created
 - 조회
 - GET
 - /v1/board/item/{idx}
 - 목록
 - GET
 - /v1/board/list/{page}
 - 수정
 - PUT
 - /v1/board/item/{idx}
 - PATCH
 - /v1/board/item/title/{idx}

- /v1/board/item/body/{idx}
- 삭제
 - DELETE
 - /v1/board/item/{idx}