

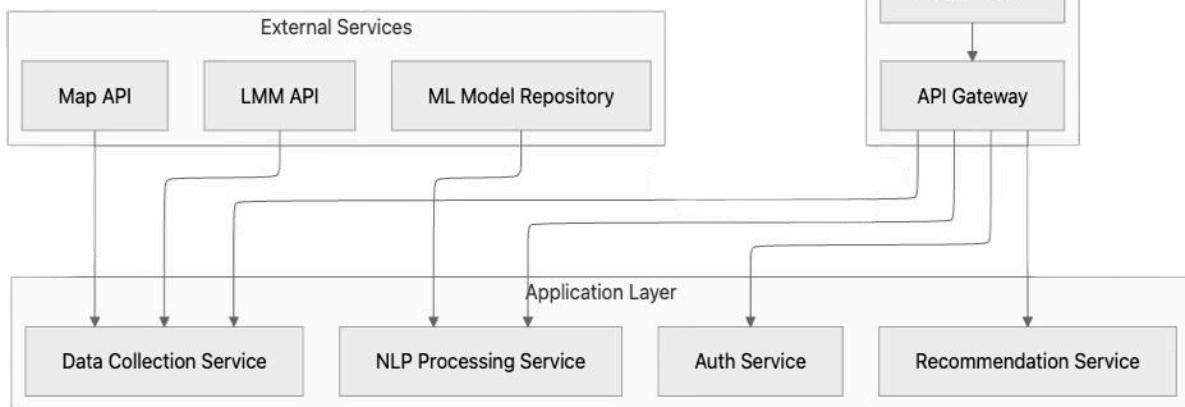
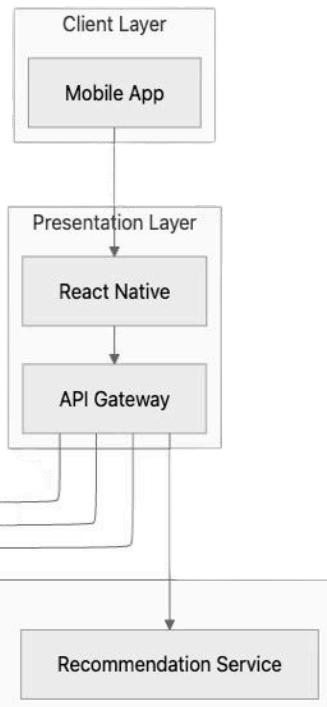
Cureat

개인화 미식 탐색 및 추천 플랫폼

시스템 아키텍처

각 핵심 기능들을 모듈화하여 시스템의 유연성 및 확장성을 확보

- API 게이트웨이를 통한 사용자 요청 분산 처리
- 백엔드 레이어에 구현된 캐싱 메커니즘을 이용하여 응답 시스템의 고효율화
- 데이터 수집 및 임베딩 생성과 같은 I/O-Bound 작업은 비동기 데이터 처리
- Vector DB 활용하여 AI 기반의 개인화 미식 추천 시스템을 구현



핵심 기능

AI 기반 자연어 분석 및 추천 시스템

사용자로부터 입력된 자연어 검색어를 Okt/ko-BERT를 통해 토큰화/정제/벡터화

외부 API 및 웹 크롤링을 통해 수집된 맛집 정보와 리뷰 텍스트를 정제/임베딩하여 Vector DB에 저장

사용자 입력/저장된 문서 벡터 간의 코사인 유사도 기반으로 각 사용자에게 개인화된 미식 추천 리스트를 제공

효율적인 데이터 수집 및 관리 시스템

외부 RESTful API 연동 및 Web Crawling 또는 RSS Feeds를 통한 실시간 정보 수집 로직으로 최신 데이터 확보

수집된 데이터에서 광고성 콘텐츠를 식별하고 필터링하는 로직으로 데이터의 품질/신뢰도 향상

사용자 경험 중심의 기능 구현

Map Service를 구현하여 사용자 위치 기반의 장소 검색 및 지리 정보 제공하는 위치 기반 검색 기능 구현

Auth Service 및 User Profile Service를 통해 사용자 인증 및 프로필 데이터 CRUD 기능 구현



문제 해결

기술적 문제 정의는 LLM 통신 및 복잡한 임베딩 조회 과정이 주요 I/O 바운드 작업으로 작용하여, 초기 추천 시스템의 API 응답 지연 (Latency)이 발생하였습니다.

이를 해결하기 위해, Python FastAPI 프레임워크의 비동기 기능을 활용하고 asyncio 라이브러리를 통해 LLM API 호출 및 Vector DB I/O 바운드 작업을 병렬적으로 처리하도록 로직을 재구성하여 동시성을 개선하였습니다. 또한, 서비스 특성상의 유사 입력에 대한 불필요한 재연산 최소화를 위해 Vector DB를 임베딩 데이터의 캐시 저장소로 활용하는 전략을 도입하였습니다.

회고

Front-end 및 Back-end 개발을 포함하여 전체 시스템을 설계부터 구현까지 주도하며 통합적인 Full-Stack 개발 역량을 확보하였습니다. 특히, Jira를 활용하여 Agile/Scrum 방식으로 프로젝트 일정을 관리하고 Git/GitHub를 통한 적극적인 협업 및 버전 관리를 경험했습니다. 이를 통해, 다양한 기술 스택을 다루고 복잡한 시스템의 안정성 문제를 해결하는 경험은 향후 솔루션 엔지니어 또는 통합 아키텍트로 성장하는 데 필요한 기반 지식과 문제 해결 능력을 확립하였습니다.



FastAPI



Hugging Face

