

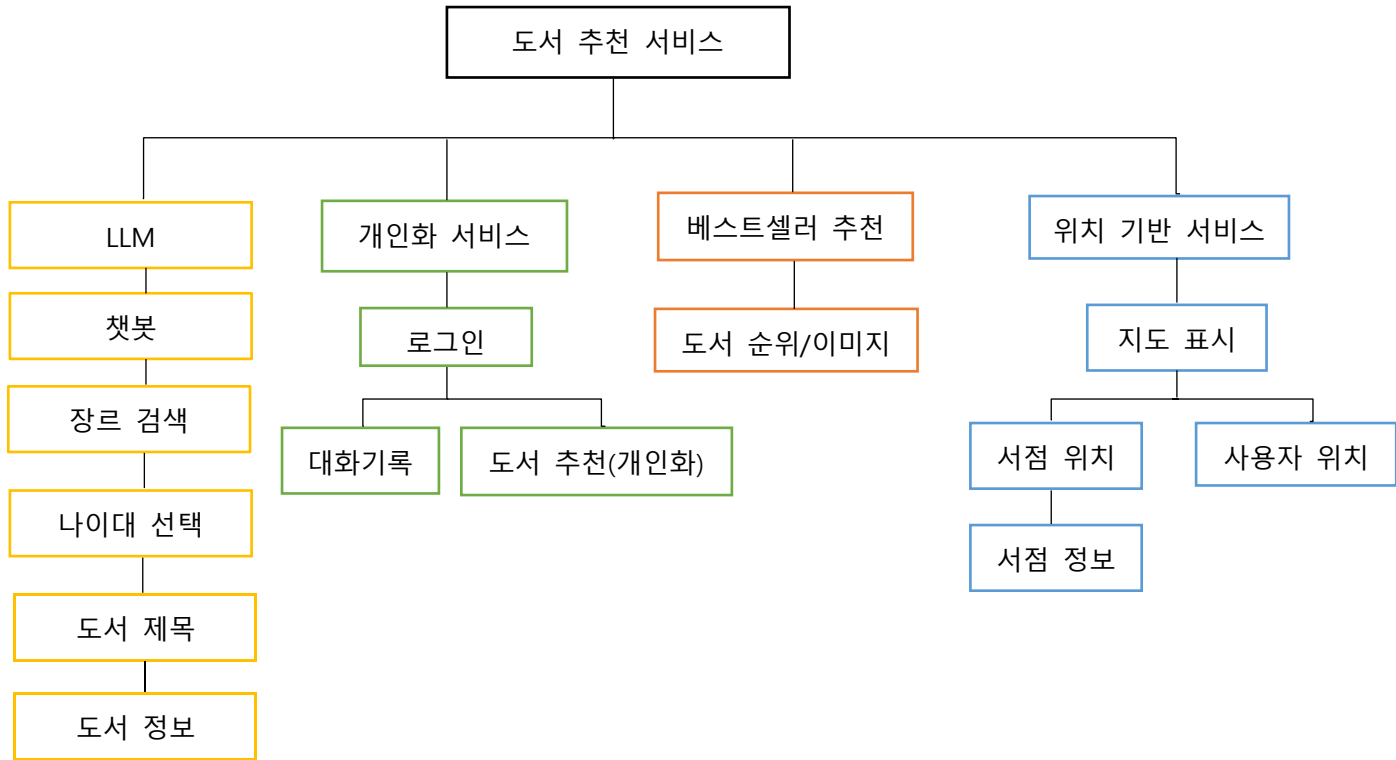
프로젝트 계획서 - 4팀

1. 프로젝트 개요

1.1 프로젝트 이름
Book4U
1.2 문제정의 문
사용자의 연령, 독서 취향 등 다양한 개인적 특성을 반영하여 선호하는 장르와 주제를 기반으로 도서를 제안
1.3 프로젝트 배경 및 필요성
<p>-프로젝트 배경-</p> <p>최근 몇 년간 인공지능(AI)과 머신러닝(ML) 기술의 발전으로 인해, 다양한 분야에서 사용자 맞춤형 서비스가 활성화되고 있습니다. 특히 도서 추천 분야에서는 사용자의 관심사, 나이, 취향에 맞는 도서를 추천 받는 것이 중요해졌습니다. 하지만, 기존의 도서 추천 서비스들은 사용자의 관심사, 나이, 취향에 맞춘 개인화된 추천에 한계가 있으며, 특히 특정 장르에 대한 세분화된 추천 서비스의 부재가 큰 문제로 지적되고 있기에, 사용자 맞춤형 도서 추천 서비스를 기획하게 되었습니다.</p> <p>-프로젝트 필요성-</p> <p>다양한 나이대와 취향을 반영한 맞춤형 도서 추천 시스템은 독서 문화 활성화에 중요한 역할을 할 수 있습니다. 기존의 도서 추천 서비스는 제한된 데이터를 기반으로 하기 때문에, 사용자 개인의 나이, 취향, 관심 장르를 정확히 반영하는 데 한계가 있습니다. 그러나 LLM 기술을 활용한 도서 추천 시스템은 이러한 한계를 극복할 수 있습니다. LLM은 방대한 양의 텍스트 데이터를 분석하고, 사용자의 대화 패턴과 선호도를 깊이 이해하여 보다 개인화된 도서 추천을 제공할 수 있습니다. LLM은 사용자가 명시적으로 언급하지 않은 취향이나 관심사를 파악하는 능력을 갖추고 있으며, 이를 통해 독자들은 자신에게 딱 맞는 책을 더 쉽게 찾을 수 있게 되며, 도서 선택의 폭이 크게 넓어집니다. LLM 기반의 추천 시스템은 사용자의 관심사를 지속적으로 반영하며, 독자가 새로운 도서와 장르를 발견하고 흥미를 잃지 않도록 도와줍니다. 이런 시스템은 전 세대에 걸쳐 독서 참여를 증진시키고, 독서 문화의 활성화에 기여할 수 있는 중요한 도구가 될 것입니다.</p>

2. 기능 구조도(Functional Decomposition Diagram)

2.1 기능 구조도



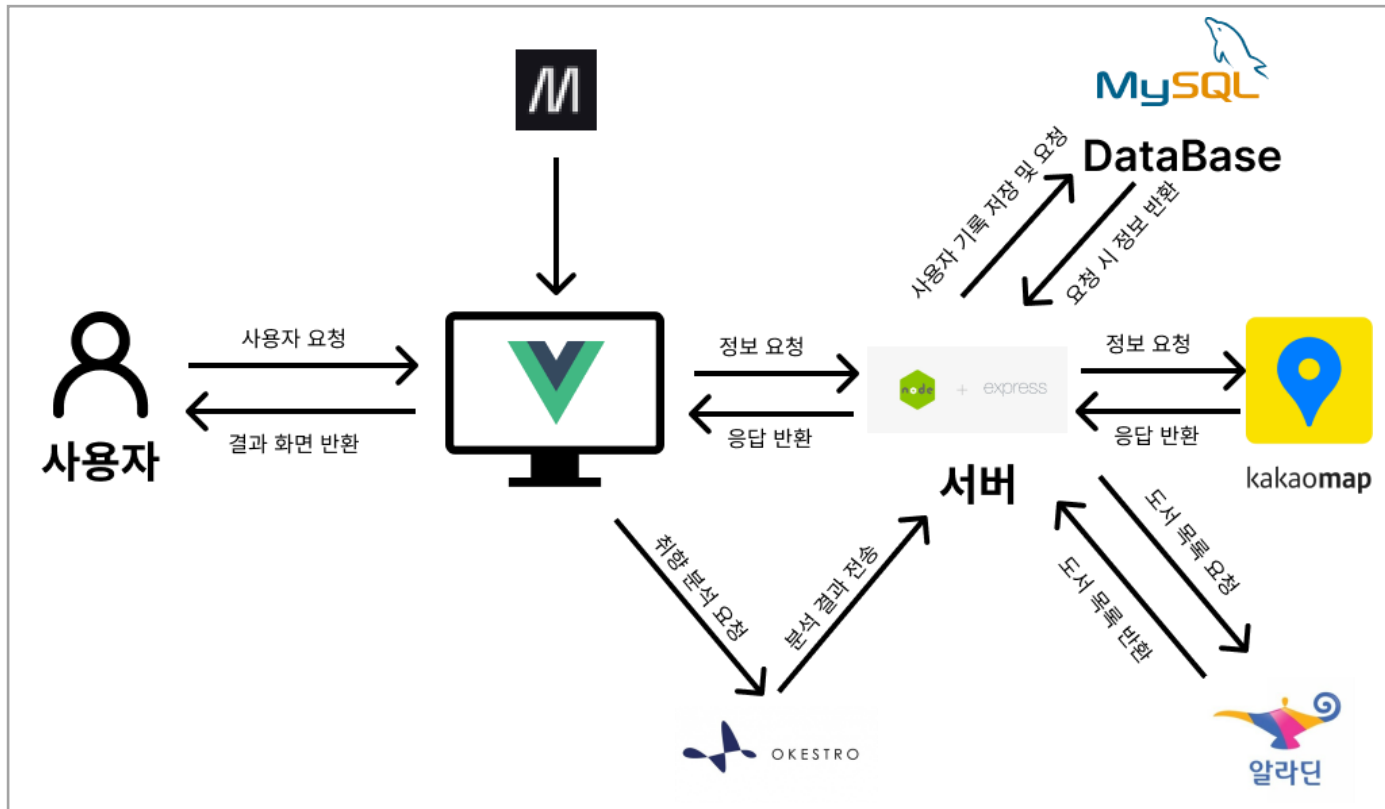
2.2 기능 설명 - 각 기능 간단 설명

2.2.1 LLM
<ul style="list-style-type: none"> - 챗봇: 질문 가능한 Prompt 형식, 장르 추천 질문 - 장르 검색: 오케스트로 LLM API 사용, 사용자가 특정 장르를 검색 - 나이대 선택: 나이대를 표시하는 체크박스 출력, 해당하는 나이대 선택 - 도서 제목: 알라딘 API 사용, 도서 제목 클릭 시 도서 정보 출력 -> (이미지, 상품명, 평점, 장르, 저자, 출판사, 출간일, 페이지 수, 정가, 줄거리)
2.2.2 개인화 서비스
<ul style="list-style-type: none"> - 로그인: 데이터베이스 사용, 사용자 개개인의 정보, 대화내용을 저장 - 대화기록: 로그인 정보를 통해 사용자 각각의 대화기록 저장, 출력 가능 - 도서 추천개인화: 사용자 조회 도서 정보와 추천도서 목록을 기반으로 도서 추천
2.2.3 베스트셀러 추천
<ul style="list-style-type: none"> - 도서 순위/이미지: 알라딘 API 사용, 주간 베스트셀러 정보를 얻어와 순위와 이미지 출력
2.2.4 서점 위치 제공
<ul style="list-style-type: none"> - 지도 표시: 카카오맵 API를 통해 지도를 구현 - 사용자 위치: Geolocation API를 사용, 현재 위치를 중심으로 설정, 표시

- 서점 위치: 현재 위치 근처의 서점을 검색, 표시
- 서점 정보: 오버레이를 통해 서점의 정보를 출력

3. 시스템 구조도

3.1 시스템 구조도



3.2 시스템 구성 요소 설명

구성요소	설명	개발 환경 및 구현 방법
DB	회원 정보 관리	MySQL DBMS를 통해, 회원 정보 관리 데이터베이스 구축 예정
서버	API 통합 및 데이터 처리	Node.js를 기반으로 하고, Express.js 프레임워크를 사용하여 서버 구현. 카카오맵 API(지도 구현), 알라딘 API(도서 정보), 오케스트로 LLM API(챗봇 시스템)를 통합하고, DB와 연동하여 개인의 데이터를 관리.
프론트 프레임워크 및 UI	사용자 인터페이스 및 상호작용	Vue.js를 기반으로 UI/UX 구현.

		Vuetify를 사용하여 일관된 디자인과 반응형 컴포넌트 구현. Geolocation API를 사용하여, 현재 위치 정보 획득. 서버로부터 받은 API 관련 정보들 화면에 출력
개발 환경 및 빌드 도구	프로젝트 구조, 개발 서버, 빌드 프로세스	Vite를 사용하여, 프로젝트 구조화 및 개발 서버 구동. 라우팅은 Vue Router를 통해 구현

4. 일정 계획

주차	마일스톤/작업	설명
5주 차	프로젝트 계획서 작성	프로젝트 목표 설정 및 팀 역할 배정
6주 차	웹 페이지 디자인(피그마)	UI/UX 디자인
7주 차	기본 구조 개발 (FE & BE)	FE/BE 기본 구조 구현, API 설계 및 초기 구현
8주 차	주요 기능 개발	FE/BE 주요 기능 구현, API 연동 및 단위 테스트
9주 차	FE/BE 연동 & 중간 발표 준비	모듈 간 연동, 통합 테스트 및 오류 수정
10주 차	중간 발표 및 피드백	개발한 기능을 토대로 발표 및 피드백
11주 차	UI/UX 개선 및 성능 최적화	피드백 기반 UI/UX 개선, 성능 최적화 작업
12주 차	추가 기능 구현 및 버그 수정	피드백 기반 추가 기능 구현, 알려진 버그 수정
13주 차	최종 테스트 및 버그 수정	전체 시스템 테스트, 개선 사항 점검
14주 차	최종 결과물 및 결과 발표	최종 결과물 시연 및 프로젝트 결과 발표

5. 역할 분담

5.1 김선준	-UI/UX 설계 및 구현- Vue.js를 사용한 사용자 인터페이스 개발, Vuetify를 활용한 반응형 디자인 구현, 프론트엔드 성능 최적화
5.2 강경민	-기능 구현 및 API 연동- 백엔드 API와의 통신 로직 구현, 상태 관리 및 데이터 흐름 설계, 프론트엔드 단위 테스트 및 통합 테스트 수행
5.3 문기엽	-서버 개발 및 관리- Node.js 및 Express.js를 사용한 서버 구축, 외부 API (카카오맵, 알라딘, 오케스트로 LLM 연동)
5.4 이희찬	-데이터베이스 설계 및 관리- MySQL 데이터베이스 설계 및 구현, 데이터 모델링 및 스키마 최적화, 데이터베이스 쿼리 최적화, 백엔드 단위 테스트 및 통합 테스트 지원