

심로그

SimLog 기술 스택 선정 이유 📖

🎯 프로젝트 요구사항 분석

핵심 요구사항

- **AI 기반 감정 분석:** 일기 텍스트를 실시간으로 분석하여 감정 상태 파악
 - **크로스 플랫폼:** Android, iOS, Web 동시 지원
 - **실시간 처리:** 사용자 입력에 대한 즉각적인 AI 분석 응답
 - **데이터 보안:** 민감한 감정 데이터의 안전한 저장 및 처리
 - **확장성:** 향후 기관별 커스터마이징 및 B2B 서비스 확장
 - **비용 효율성:** 스타트업 단계에서의 경제적 운영
-

🏗️ Backend 기술 스택

FastAPI 0.115.13 (Python)

선택 이유:

- **비동기 처리:** AI API 호출과 같은 I/O 집약적 작업에 최적화
- **자동 API 문서화:** Swagger UI를 통한 실시간 API 문서 생성 및 테스트
- **타입 힌팅:** Python의 타입 힌팅을 활용한 코드 안정성 확보
- **성능:** Node.js 수준의 고성능 웹 프레임워크
- **개발 속도:** Pydantic 모델과 자동 검증으로 빠른 개발 가능

SimLog에서의 활용:

- 일기 텍스트를 받아 OpenAI API로 실시간 감정 분석
- JWT 토큰 기반 사용자 인증 처리
- 주간/월간 감정 요약 데이터 생성 및 캐싱

MySQL 8.0 (Railway)

선택 이유:

- **관계형 데이터베이스:** 사용자-일기-분석결과 간의 복잡한 관계 처리
- **JSON 컬럼 지원:** AI 분석 결과의 유연한 저장 (감정 분석 JSON, 키워드 배열)
- **트랜잭션 안정성:** 일기 작성과 AI 분석 결과 저장의 원자성 보장
- **Railway 통합:** 백엔드와 동일 플랫폼에서 관리로 운영 효율성

SimLog에서의 활용:

- 사용자 정보, 일기 내용, AI 분석 결과 저장
- 주간 요약 캐시, 공유 리포트, 정원 아이템 관리
- JSON 형태로 감정 분석 결과의 세부 데이터 저장

SQLAlchemy 2.0.0

선택 이유:

- **Python 생태계 표준:** 가장 널리 사용되는 Python ORM
- **타입 안전성:** Python 타입 힌팅과 완벽 호환
- **마이그레이션 지원:** Alembic을 통한 데이터베이스 스키마 관리
- **성능 최적화:** 지연 로딩, 즉시 로딩 등 다양한 로딩 전략

SimLog에서의 활용:

- User, Record, GardenItem 등 8개 모델의 관계 정의
- 복잡한 쿼리 (주간 감정 추이, 연속 출석 등) 처리
- JSON 컬럼을 활용한 AI 분석 결과 저장

JWT (JSON Web Token)

선택 이유:

- **서버리스 아키텍처 적합:** 서버 상태 없이 클라이언트에서 토큰 관리
- **확장성:** 마이크로서비스 환경에서 사용자 인증 정보 공유
- **보안:** 서명된 토큰으로 무결성 보장
- **성능:** 데이터베이스 조회 없이 토큰 검증 가능

SimLog에서의 활용:

- 모바일 앱과 백엔드 간 안전한 인증

- 토큰 만료 시간 설정으로 보안 강화
 - 사용자별 권한 관리 (일반 사용자 vs 개발자)
-

Frontend 기술 스택

Flutter 3.16.5 (Dart 3.2.3)

선택 이유:

- **단일 코드베이스:** Android, iOS, Web 동시 개발로 개발 효율성 극대화
- **네이티브 성능:** 각 플랫폼의 네이티브 컴포넌트 활용
- **Hot Reload:** 실시간 코드 변경 반영으로 빠른 개발
- **풍부한 위젯:** 감정 색상 표현, 애니메이션 등 UI 구현 용이
- **Google 지원:** 지속적인 업데이트와 안정성 보장

SimLog에서의 활용:

- 감정 색상을 시각적으로 표현하는 커스텀 위젯
- 마음 정원의 드래그 앤 드롭 인터페이스
- 크로스 플랫폼 일관된 사용자 경험 제공

Flutter 기본 상태 관리

선택 이유:

- **프로젝트 규모 적합:** 복잡한 상태 관리 라이브러리 없이도 충분
- **학습 곡선:** 팀원들이 쉽게 이해하고 유지보수 가능
- **성능:** 불필요한 리빌드 방지로 앱 성능 최적화
- **Flutter 생태계:** Flutter의 기본 패턴을 따르는 일관성

SimLog에서의 활용:

- 사용자 정보, 일기 작성 상태 관리
- 마음 정원 아이템 배치 상태 관리
- 로딩 상태와 에러 상태 처리

http package & url_launcher

선택 이유:

- **Flutter 공식 패키지:** 안정성과 지속적인 업데이트 보장
- **가벼운 크기:** 앱 크기 최적화
- **플랫폼 호환성:** Android, iOS, Web 모두에서 안정 동작
- **외부 링크 처리:** Google Forms, 상담 신청 등 외부 서비스 연동

SimLog에서의 활용:

- 백엔드 API와의 HTTP 통신
- Google Forms 상담 신청 링크 열기
- 7일 요약 공유 링크 생성 및 열기

AI & 외부 서비스

OpenAI GPT-4o mini

선택 이유:

- **감정 분석 특화:** 텍스트 기반 감정 분류에 뛰어난 성능
- **한국어 이해도:** 한국어 텍스트의 뉘앙스와 맥락 이해 우수
- **비용 효율성:** GPT-4 대비 1/10 비용으로 경제적 운영
- **응답 속도:** 빠른 응답 시간으로 실시간 분석 가능
- **안정성:** OpenAI의 안정적인 인프라와 99.9% 가용성

SimLog에서의 활용:

- 일기 텍스트의 감정 분류 (로버트 플루치크 8가지 감정)
- 감정 강도 분석 (1-10 스케일)
- 텍스트 요약 및 키워드 추출
- 분석 근거 제공으로 사용자 신뢰도 향상

Naver Clova STT

선택 이유:

- **한국어 특화:** 한국어 음성 인식 정확도 국내 최고
- **방언 및 억양:** 다양한 한국어 방언과 억양에 대한 높은 인식률
- **노이즈 처리:** 일상 환경의 배경 소음에 강한 인식 성능

- **국내 서비스:** 네트워크 지연 최소화

SimLog에서의 활용:

- 음성으로 일기 작성 기능
- 접근성 향상 (시각 장애인, 노약자 지원)
- 빠른 일기 작성으로 사용자 편의성 증대

Fallback 시스템 (키워드 기반 분석)

선택 이유:

- **서비스 안정성:** AI API 실패 시에도 기본 분석 제공
- **로버트 플루치크 이론:** 심리학적으로 검증된 감정 분류 체계
- **빠른 응답:** AI API 호출 시간 없이 즉시 분석 결과 제공
- **비용 절약:** API 호출 실패 시 재시도 비용 절약

SimLog에서의 활용:

- AI API 키 누락, 네트워크 오류, API 한도 초과 시 대체 분석
- 부정/긍정 키워드 기반 기본 감정 분류
- 사용자에게 명확한 분석 방법 안내

Google Forms

선택 이유:

- **무료 서비스:** 상담 신청 폼으로 적합한 비용 효율성
- **사용자 친화적:** 직관적인 폼 작성 인터페이스
- **응답 관리:** 상담사가 쉽게 응답을 확인하고 관리
- **커스터마이징:** 기관별 맞춤 폼 생성 가능

SimLog에서의 활용:

- 상담 신청 폼으로 사용
- 기관별 맞춤 폼으로 확장 가능
- 응답 데이터 수집 및 관리

배포 및 인프라

Railway

선택 이유:

- **개발자 친화적:** GitHub 연동으로 자동 배포
- **통합 관리:** 백엔드와 데이터베이스를 한 플랫폼에서 관리
- **자동 HTTPS:** SSL 인증서 자동 발급 및 관리
- **스케일링:** 트래픽 증가에 따른 자동 스케일링
- **비용 효율성:** 사용량 기반 과금으로 초기 비용 절약

SimLog에서의 활용:

- FastAPI 백엔드 자동 배포
 - MySQL 데이터베이스 관리
 - 환경 변수 관리 (.env 파일)
 - 로그 모니터링 및 에러 추적
-

개발 도구

Git & GitHub

선택 이유:

- **분산 버전 관리:** 팀 협업과 코드 히스토리 관리
- **브랜치 전략:** 기능별 개발과 안전한 병합
- **GitHub 연동:** Railway 자동 배포와 연동
- **코드 리뷰:** Pull Request를 통한 품질 관리

Postman

선택 이유:

- **API 테스트:** 백엔드 API 엔드포인트 테스트
- **팀 협업:** API 문서 공유 및 테스트 케이스 관리
- **자동화:** API 테스트 자동화로 품질 보장
- **디버깅:** 요청/응답 상세 분석

VS Code / Cursor

선택 이유:

- **AI 코딩 지원:** GitHub Copilot, Cursor AI 등으로 개발 효율성 증대
 - **플러그인 생태계:** Python, Flutter, Git 등 다양한 언어/도구 지원
 - **무료 사용:** 개인/팀 모두 무료로 사용 가능
 - **통합 터미널:** 개발 환경 통합 관리
-

확장성 고려사항

모듈화된 구조

- **API 기반:** 다양한 플랫폼과의 연동 용이
- **마이크로서비스 준비:** 향후 서비스 분리 가능한 구조
- **데이터 독립성:** 기관별 데이터 분리 저장 가능

기술적 확장성

- **다양한 AI 모델:** GPT 외 다른 AI 모델로 교체 가능
- **다중 언어 지원:** 한국어 외 다른 언어 확장 가능
- **플랫폼 확장:** 데스크톱, 웨어러블 등 추가 플랫폼 지원

비즈니스 확장성

- **B2B 서비스:** 기업/기관 대상 라이선스 판매
 - **화이트라벨:** 기관별 브랜딩 적용 가능
 - **API 서비스:** 타사 서비스에 감정 분석 API 제공
-

기술 선택의 핵심 원칙

1. **비용 효율성:** 스타트업 단계에서 경제적 운영
2. **개발 속도:** 빠른 프로토타이핑과 MVP 개발
3. **확장성:** 향후 성장을 고려한 기술 선택
4. **안정성:** 프로덕션 환경에서의 안정적 운영
5. **사용자 경험:** 직관적이고 반응성 좋은 인터페이스
6. **보안:** 민감한 감정 데이터의 안전한 처리

이러한 기술 스택 선택을 통해 SimLog는 현재의 부산대학교 서비스를 넘어 다양한 기관과의 연계를 통한 확장 가능한 마음 건강 관리 플랫폼으로 발전할 수 있는 기반을 마련했습니다.