

졸업프로젝트

---

# 과제기획서 발표

LOG MATE

---

팀원 : 강찬욱, 양승원, 정주연

2025.05.17

---

## Content

# 목차

1

## 사업 개요

- 1-1. 동기 및 기대효과
- 1-2. 사업 내용 및 특성

3

## 향후 추진 계획

- 3-1. 사업화 추진내용
- 3-2. 사업계획 차질시 대처방안

2

## 시장 분석

- 2-1. 목표시장 규모 및 동향
- 2-2. 사업화 가능성 및 마케팅 전략
- 2-3. 고객발굴 및 검증 계획
- 2-4. 성장가능성

4

## 일정 및 소요자원/조달 계획

- 4-1. 추진일정
- 4-2. 소요자원/조달계획

Part 1

# 사업 개요

### 1. 동기

- 진입 장벽 해소
  - 기존 로그 솔루션은 복잡한 설치와 전문가용 설정 필요
  - 학습 비용이 큼
- 기술 접근성 확대
  - 개발 초심자도 쉽게 로그 수집 가능
  - Web UI 를 통한 쉬운 설정 가능

### 1. 동기

- 문제 인지부터 대응까지 자동화
  - 수집, 시각화, 이상 탐지를 하나의 솔루션으로 통합
  - 누구나 쉽게 쓸 수 있는 DevOps 도구 구현 목표

## 2. 기대 효과

- 초기 구축 비용 및 인력 리소스 절감
- AI 기반의 이상 탐지를 통한 사용자의 능동적인 대응 환경 제공
- 사용자 친화적 UI로 기술 접근성 향상
- 로그 관리의 효율성과 시장 경쟁력 강화

## 1. 분야

- DevOps 및 모니터링 분야의 SaaS 플랫폼

## 2. 특징

- 간단한 환경 구축
- 실시간 로그 수집
- AI 기반 분석 기능
- 초기 스타트업, 소규모 개발팀, 1인 개발자도 쉽게 사용가능한 경량 모니터링 서비스

### 3. 기술 개발 내용

- 에이전트 서버 (로그 수집기)
  - 실시간 로그 수집, 파싱, 필터링, 전송 지원
  - 경량 실행 파일로 설정 없이 실행 가능
  - 동적 설정 변경 지원

### 3. 기술 개발 내용

- 스트리밍 서버
  - 로그 데이터 영구 저장 지원
  - 대용량 로그 데이터 수집 지원
  - Kafka Consumer + WebSocket + WebFlux 기반 비동기 구조로 대용량 요청 처리 지원

### 3. 기술 개발 내용

- API 서버
  - 사용자 인증, 대시보드 정보, Webhook 알림 처리 등  
REST API 제공
- 프론트엔드 서버
  - Vue.js 기반 실시간 로그 뷰어, 설정 마법사 등 사용자  
UI 제공

Part 2

## 시장 분석

### 1. 소비자 분석

- 타겟 고객
  - 1인 개발자, 초기 스타트업, 소규모 개발 조직
- 공통 특징
  - 클라우드 기반 운영, 인프라 운영 리소스 부족, 기존 시스템의 진입 장벽 부담
- 관심사
  - 빠른 도입 & 쉬운 사용, 낮은 학습 곡선, 저비용 대안 선호

## 2. 시장 규모 및 동향

- 국내 서버/로그 관리 도구 수요
  - 클라우드 확산과 SaaS 기반 서비스 증가로 실시간 운영 모니터링 도구 수요 증가
  - Seed~Series A 스타트업 수만 1,000개 이상, 대부분 전문 DevOps 인력 없이 운영

## 2. 시장 규모 및 동향

- 기술 동향
  - 대규모 플랫폼은 Kafka 기반 스트리밍 사용 중
  - AI 기술과 외부 메신저 연동 기능이 중요시됨
  - 간단한 설정, 즉시 사용가능 등 편의성이 SaaS 시장의 주요 구매 요인
- 글로벌 로그 모니터링 및 분석 시장 규모
  - 24년 기준 약 18억 달러 규모, 연평균 12% 이상 성장 중
  - Datadog, Elastic 등이 선도하고 있으나 대부분 엔터프라이즈 중심

### 3. 경쟁자 분석

항목	Loggly	Datadog	ELK Stack	Grafana Loki	<b>LogMate</b>
설정 방식	스크립트 기반	YAML 설정 파일	다단계 구성 파일	직접 YAML 작성	웹 UI 설정 주입
초기 필수 설정 항 목 수	2~3개	5~6개	수십 개	5~10개	<b>1개 (수집 파일 경로)</b>
고급 설정 범위	제한적 (Syslog)	다양한 기능 (regex, 알림)	매우 복잡 (DSL 필요)	제한적 (json/regex)	<b>UI 기반 고급 옵션 제공 (필터, 파서, 알림 등)</b>
설정 변경 방식	일부 반영	재시작 필요	재시작 필요	재시작 필요	<b>HTTP 호출로 실시간 반영</b>
대시보드	SaaS 제공	SaaS 제공	Kibana 별도	Grafana 별도	<b>SaaS 제공</b>
AI 분석 기능	없음	이상탐지 지원	없음	없음	<b>이상탐지 지원</b>

### 3. 경쟁자 분석

- 차별점
  - 진입은 간편하게, 확장은 유연하도록 설계
  - 고급 설정 및 기능은 사용자 친화적 UI로 제공
  - 초보자는 기본 설정으로 수집
  - 숙련자는 고급 기능 조작
  - 기존 도구들은 Yaml, DSL 등 높은 진입 장벽 요구
  - LogMate는 필요한 기능만 UI를 통해 직관적으로 조작 가능하도록 만드는 것에 집중

## 4. SWOT 분석

- Strengths
  - 무설정 설치 및 동적 설정 주입 기능
  - 초보자도 쉽게 사용할 수 있는 UI 중심 구성
  - WebSocket 기반 실시간 로그 스트리밍
  - 내장형 AI 이상 탐지 및 Slack/WebHook 연동
- Weaknesses
  - 대규모 엔터프라이즈 환경에 대한 확장성은 제한적
  - 인프라 복잡도 대비 기능 범위는 간결한 편

## 4. SWOT 분석

- Opportunities
  - SaaS 도입 수요 증가
  - DevOps 인력이 없는 소규모 팀의 운영 자동화 요구 증가
  - 간편하고 실용적인 도구에 대한 시장 선호도 상승
- Threats
  - 기존 대형 SaaS 기업의 가격 인하 또는 간편 서비스 출시
  - 오픈소스 로그 도구 학습 커뮤니티의 성장으로 진입장벽 감소

### 1. 사업화 가능성

- 통합 플랫폼
  - 로그 수집, 분석, 시각화, 알림을 하나의 플랫폼에서 제공
  - 복잡한 설정, 설치 없이도 운영 가능
- 비즈니스 모델
  - ‘오픈소스 Agent’ + ‘유료 클라우드 대시보드’
  - 프리미엄 SaaS 모델로 확장 용이
  - 고객 확보와 수익화 전략을 병행 가능

## 2. 마케팅 전략

- 타겟 중심 확산 전략
  - 1인 개발자 / 소규모 개발팀 대상
    - 무료 플랜 + 오픈소스 Agent
    - 바텀업 확산 & 개발자 주도 도입 유도
- 인바운드 마케팅 & 커뮤니티 중심
  - Slack / Discord 등 외부 메신저 연동
  - velog, HackerNews 등 Dev 커뮤니티 홍보

## 2. 마케팅 전략

- 체험 기반 확산 & 파트너십
  - 기술 밋업, 해커톤, 클라우드 행사 참가
  - 시연 + 부스 운영

### 3. 수익화 방안

- 기본 무료 플랜 제공
  - Agent 서버는 무료 배포 (GitHub 오픈 소스)
  - 로그 스트리밍 기능 기본 무료 제공
- 유료 플랜 전환 요소
  - 팀 워크스페이스 사용자 수 확장 및 권한 설정 기능
  - AI 이상 탐지 및 주간 분석 알람 기능
  - Webhook 연동 개수 확장
  - 장기 로그 저장

### 3. 수익화 방안

- 확장형 수익 모델
  - OEM 라이선스 제공
  - Slack / Discord 연동 유료 플랜
- 수익화 목표
  - 경량, 실시간, 확장성을 유지하며 점진적 유료화

### 1. 목표고객군

- 로그 수집 및 분석 인프라에 익숙하지 않은 개발자
- 시간과 자원이 부족한 1인 개발자
- 소규모 스타트업 개발팀
- 개발 입문자

### 2. 고객 발굴

- 개발자 커뮤니티, Tistory/Velog 블로거, GitHub, 인프런 등
- 사이드 프로젝트 및 개인 서버를 운영 중인 개발자 중심으로 설문 및 피드백 수립
- 간단한 MVP 체험판 제공을 통해 사용 가능성과 문제점 도출

### 3. 검증 계획

- 가설 수립
  - 간단한 로그 수집 및 분석 기능에 대한 수요가 있을 것으로 예상
- 설문 및 인터뷰를 통해 정성적, 정량적 데이터 수집
  - “어떤 방식으로 로그 수집하는지”, “로그 수집 중 어려운 점은?”, “AI 기능이 있다면 사용할 의향 있는지”

### 3. 검증 계획

- MVP 배포 및 피드백 수집
  - 대시보드 시각화 가능, 이상탐지 등
- 정량 분석
  - 기능별 사용률, 반응률, 재방문율 등을 통해 니즈 검증

## 1. 개발 제품/ 서비스군의 수명주기 및 지속성장 전략

- 초기 단계
  - 실시간 로그 수집, 웹 UI 설정, 이상탐지 구현
  - 설정 없는 간편 설치로 입문자 유입 확보
- 성장 단계
  - Slack/Discord 알림 연동, 다중 로그/멀티라인 처리
  - 개발팀 내 도입 확산 및 팀 단위 유료 전환

## 1. 개발 제품/ 서비스군의 수명주기 및 지속성장 전략

- 고도화 단계
  - 사용자 맞춤 대시보드, AI 기반 사용자 행동 분석 구현
  - 엔터프라이즈 기능 확장 및 리포트 기반 서비스화
- 확산 단계
  - 타 SaaS 및 클라우드 인프라와 연동을 통한 시장 확장

Part 3

## **향후 추진 계획**

## 1. 개발목표, 개발인력 구성

- 주요 기능
  - 간단한 구성과 설치 환경에서의 로그 수집
  - AI 기반 이상 탐지 및 사용자 행동 분석
  - 대시보드 시각화 및 알림 기능
  - 팀 대시보드 및 워크스페이스 기능

## 1. 개발목표, 개발인력 구성

- 주요 목표
  - 경량화된 구조를 바탕으로 로그 수집 환경의 빠른 구축과 쉬운 유지보수를 가능하게 하는 것을 목표로 함
- 개발 인력 (총 3인)
  - 백엔드 및 인프라 개발, 프론트엔드 작업 : 2인
  - AI 모델 및 분석 기능 개발, 프론트엔드 작업 : 1인

## 2. 개발 일정

- Sprint 1 - MVP 기반 구축 및 구조 정의 (3주)
  - Agent 서버
    - 설정 주입 서버 기본 구조 구현
    - 로컬 설정 불러오기, 유효성 검사 및 상태 대기
    - 로그 파일 tail 기본 동작 구현

## 2. 개발 일정

- Sprint 1 - MVP 기반 구축 및 구조 정의 (3주)
  - API 서버
    - 사용자 인증 (회원가입/로그인 API)
    - 팀 및 대시보드 CRUD API 설계 및 일부 구현
    - 설정 전송 API (POST /config) 처리 로직 기본 구현

## 2. 개발 일정

- Sprint 1 - MVP 기반 구축 및 구조 정의 (3주)
  - Streaming 서버
    - Spring WebFlux 기반 Kafka Consumer 설계
    - WebSocket 연결 수립 및 세션 관리 기능 개발
    - Kafka ↔ WebSocket 브리지 구현 MVP

## 2. 개발 일정

- Sprint 1 - MVP 기반 구축 및 구조 정의 (3주)
  - Frontend 서버
    - 로그인/회원가입 UI
    - 팀 선택 및 대시보드 리스트 UI
    - 설정 전송 기본 폼 (경로 + URL만) 구현

## 2. 개발 일정

- Sprint 1 - MVP 기반 구축 및 구조 정의 (3주)
  - AI 서버
    - 로그 이상탐지 모델 기획 및 구조 설계
    - 테스트용 단순 Rule 기반 이상탐지 함수 작성
    - Flask/FastAPI 서버 기본 템플릿 작성

## 2. 개발 일정

- Sprint 2 - 설정 관리 및 실시간 스트리밍 (3주)
  - Agent 서버
    - 설정 수신 → 로그 수집 재시작 로직
    - 다중 로그파일 수집 및 멀티라인 병합 로직
    - 파싱 및 필터링 설정 적용 처리

## 2. 개발 일정

- Sprint 2 - 설정 관리 및 실시간 스트리밍 (3주)
  - API 서버
    - 파싱/필터링 룰 저장 및 전달 API
    - 설정 등록 UI ↔ 설정 API 연동 완성
    - Webhook 알림 API 초안

## 2. 개발 일정

- Sprint 2 - 설정 관리 및 실시간 스트리밍 (3주)
  - Streaming 서버
    - 필터 조건 처리 (세션별 필터 저장 및 적용)
    - 로그 메시지 필터링 기능 구현
    - Gzip 압축 메시지 해제 처리
    - 사용자별 브로드캐스트 적용

## 2. 개발 일정

- Sprint 2 - 설정 관리 및 실시간 스트리밍 (3주)
  - Frontend 서버
    - 설정 UI 고도화 (필터, 멀티라인 등)
    - 실시간 로그 뷰어 WebSocket 연결
    - 설정 등록 → API 호출 연동 완성

## 2. 개발 일정

- Sprint 2 - 설정 관리 및 실시간 스트리밍 (3주)
  - AI 서버
    - 간단한 이상 탐지 로직 (IP 요청 패턴 등)
    - AI 탐지 결과 → API 서버로 POST 연동
    - 로그 기반 예외 케이스 샘플 구축

## 2. 개발 일정

- Sprint 3 - AI 연동 준비 + 안정성 및 알림 기능 (3주)
  - Agent 서버
    - 수집 로그 UUID 부여 및 중복 방지 로직
    - 로그 전송 실패 시 재시도 버퍼 처리
    - 선택적 압축 전송 및 HTTPS 연동

## 2. 개발 일정

- Sprint 3 - AI 연동 준비 + 안정성 및 알림 기능 (3주)
  - API 서버
    - Slack, Discord Webhook 전송 기능
    - 알림 이벤트 로그 기록 API
    - AI 분석 요청 API 스펙 정의

## 2. 개발 일정

- Sprint 3 - AI 연동 준비 + 안정성 및 알림 기능 (3주)
  - Streaming 서버
    - Kafka 장애 대응 (백오프, 재연결)
    - Elasticsearch/MySQL 저장 기능 구현
    - AI 서버에 이상탐지용 로그 전송 처리

## 2. 개발 일정

- Sprint 3 - AI 연동 준비 + 안정성 및 알림 기능 (3주)
  - Frontend 서버
    - 알림 설정 UI 및 전송 시각화
    - 마이페이지 및 팀 멤버 관리 UI
    - 실시간 로그 필터링 UI 완성

## 2. 개발 일정

- Sprint 3 - AI 연동 준비 + 안정성 및 알림 기능 (3주)
  - AI 서버
    - AI 모델 버전 관리 및 재학습 구조 설계
    - 상태코드/행동 기반 탐지 시각화 제공
    - 비정상 로그 알림/시각화 응답 API 구축

### 1. 위험요소 분석

- 기술 개발 지연
  - Kafka, WebSocket, 비동기 처리 등 시스템 구현 복잡도에 따른 개발 지연
- 인력 확보 및 유지
  - 핵심 개발 인력 부족
- 서버 운영 비용
  - 실시간 처리와 분리된 여러개의 클라우드 서버에 대한 과금 증가

## 1. 위험요소 분석

- 마케팅 반응 부진
  - 커뮤니티 중심 마케팅 효과 미흡 가능성
- 시장 경쟁 심화
  - 유사 SaaS 서비스 출시 및 대기업 시장 유입 가능성

## 2. 대응 전략

- 기술 개발
  - 모듈 분리 및 핵심 기능 우선 구현
  - 오픈소스 적극 활용
- 인프라 운영
  - 프리티어 우선 활용
  - 모듈형 클라우드 구조 사용

## 2. 대응 전략

- 마케팅 반응 부진
  - 블로그, Velog, 유튜브 채널 확산
- 시장 경쟁 심화
  - 스타트업 전용 시장 집중
  - 손쉬운 설정, 사용자 친화적 UI 등 차별화 유지
  - B2B 제휴 추진

Part 4

## 일정 및 소요자원/조달 계획

# 추진 일정

---

4-1. 추진일정			각 세부 과업별로 일정 추진계획 수립							
내용	2025년도 8분기								비고	
	1/8	2/8	3/8	4/8	5/8	6/8	7/8	8/8		
개발 환경 구축 및 기본 환경 설정	●	●							로그 수집	
다중 사용자 테스트		●	●	●	●	●			다중사용자 대응	
대시보드 UI 개선			●	●	●	●	●	●	사용자기능 개선	
AI 이상탐지 모델				●	●				웹 로그 기반	
모델과 기능 통합					●	●			통합 및 점검	
중간점검 및 피드백						●	●		오류 수정	
최종 수정 및 개발완료							●	●	개발완료	
최종 발표 및 문서화							●	●	발표	

## 1. 인적 자원

- 개발 인력 총 3명
  - 백엔드 및 인프라 개발 2명
  - AI 모델 및 분석 기능 개발 1명
  - 프론트엔드 작업은 3명의 개발자가 역할을 분담하여 개발 예정
- 역할에 맞춰 애자일 방식으로 주기적으로 점검 및 기능 구현 예정

## 2. 기술 자원

- 개발 언어 및 프레임워크
  - JAVA, Spring, Vue.js
- 로그 처리
  - Kafka, Elastic Search, Spring WebFlux

## 2. 기술 자원

- 인프라 및 데이터베이스
  - AWS, MySQL
- AI 모델
  - 웹 로그 기반 이상 요청을 감지하는 AI 모델 적용
  - Isolation Forest 등

### 3. 개발 장비 및 인프라

- 개인 노트북을 주 개발 환경으로 사용
- GitHub, AWS 클라우드 등 유료/공개 리소스 활용
- 실사용 테스트는 로컬 및 팀원 간의 테스트 환경을 통해 검증

## 4. 재정 및 조달 방식

- 자금
  - 졸업 프로젝트 과제로 예산 0원
- 필요한 자원 조달
  - 개인 소유 장비 이용
  - 필요시 개인 사비를 통해 조달
- 별도의 장비나 인건비, 외부 투자 등은 계획하지 않음

# 감사합니다

---

양식 출처: 2025 국방기술을 활용한 창업 경진대회 사업계획서

대회 홈페이지 링크: <http://www.dapa-startup.kr/?c=20/21>

사업계획서 링크:

<http://www.dapa-startup.kr/2025%EC%82%AC%EC%97%85%EA%B3%84%ED%9A%8D%EC%84%9C.hwp>