

Lili Agent 베타 MVP 중간 정리

작성일: 2025-11-28 상태: 논의 중 (미확정 사항 있음) 관련 Jira: PS-651

1. 제품 개요

기본 정보

- 제품명: Lili Agent
- 본질: Claude Code Desktop Wrapper (Electron 기반 GUI)
- 비전: "개발자가 VS Code를 설치하고 개발을 시작하듯, 비개발자가 이 앱을 설치하고 AI Agent로 업무를 시작한다"

타겟 사용자

- 마케터, 기획자, 디자이너
- 리서처, 영업, CS, 재무, 인사
- 경영진/의사결정자
- 공통점: 비개발자

핵심 차별화 포인트

"ChatGPT의 쉬운 UI + Claude Code의 강력한 기능"

경쟁 제품	비개발자 접근성	주요 문제
ChatGPT	★★★★★	파일 접근 제한, 업무 도구 연동 없음
Claude Code CLI	★	터미널 필수, GUI 없음
Cursor/Windsurf	★ ~ ★★	IDE 기반, 개발자용
Antigravity	★★	IDE 환경, 에이전트 관리 복잡

Lili의 포지션: ChatGPT처럼 쉽지만, 파일 시스템 접근 + MCP 도구 연동 가능

2. 시장 검증: Claude 비개발자 사용 사례

2.1 Claude Code CLI 활용 사례 (코딩 포함)

"문제를 설명할 수 있는 사람이라면 누구나 솔루션을 만들 수 있다"

직군	사용 사례	효과
마케터	15개 경쟁사 동시 분석, 광고 수백 개 자동 생성	시간→분 단위로 단축
법무팀	"Phone Tree" 시스템 - 적합한 변호사 연결	자체 도구 개발
콘텐츠팀	트렌드 조사 → 콘텐츠 생성 → SEO 최적화 파이프라인	주 8-10시간 절약
리서처	고객 피드백, 설문, 통화 녹취 분석 → 구조화된 DB	압도적 양의 정보 처리
HR	면접 녹취록 분석 → Notion 카드 자동 생성	채용 프로세스 자동화
데이터팀	복잡한 인프라 문제 해결, 문서화된 워크플로우	비개발자도 데이터 접근

대표 성과:

- Syncari**: 마케팅 자동화로 SQL 비율 23% 증가
- 헬스케어 플랫폼**: 6주 만에 \$1.2M 파이프라인 회복
- Obvi**: 월 10,000+ 고객 문의 자동 처리, 응답 속도 65% 향상

2.2 Claude 일반 사용 사례 (코딩 없이, ChatGPT처럼)

엔터프라이즈 기업들의 실제 활용

기업	직군	활용 방식	효과
IG Group	마케팅	다국어 콘텐츠 생성 (규제 준수)	출시 속도 3배 향상
IG Group	HR	성과 피드백 일관성 있게 작성	지역별 편차 해소
IG Group	분석	데이터 분석 자동화	주 70시간 절약
Zapier	전사	800+ 내부 Claude 에이전트 운영	작업 완료 10배 증가 (YoY)

기업	직군	활용 방식	효과
Salesforce	영업/CS	Agentforce - 고객 데이터 분석→거래 실행	완전 자동화 워크플로우
법률 사무소	법무	문서 요약, 리서치 자동화	4일→4분으로 단축

일반 업무 활용 영역:

- 회의록 → 액션 아이템 자동 추출
- 이메일 분류 및 우선순위 지정
- 스프레드시트 데이터 모델링
- 프로젝트 계획 수립
- 프레젠테이션 초안 작성
- SOP(표준운영절차) 문서화

2.3 No-Code 워크플로우 자동화 사례

제품	설명
CodeWords	비개발자가 Claude로 워크플로우 자동화 구축
Latenode	드래그앤드롭으로 Claude + CRM/이메일/메신저 연동
n8n + Claude	단순 RPA → 지능형 프로세스 자동화(IPA)
Dust	Claude 기반 기업용 AI 에이전트 플랫폼 (MCP 연동)

2.4 시장 인사이트

"Agentic coding은 기술/비기술 업무의 경계를 허물고 있다"

- Claude 시장 점유율: 3.2% (2025.4 기준, 전년 대비 +1.1%p)
- 분기별 사용자 증가율: 14% (AI 플레이어 중 최고)
- Microsoft 365 Copilot에 Claude 모델 통합 (2025)

3. 확정된 MVP 기능 (개인용 5가지)

3.1 원클릭 환경 설치

비개발자의 가장 큰 진입장벽 제거

항목	설명
Node.js 18+	자동 설치
Claude Code CLI	npm 자동 설치
환경 검증	claude doctor 연동

성공 지표: 설치 완료율 90% 이상

3.2 채팅 UI (Chat Mode)

- 메시지 버블 UI
- 마크다운 렌더링
- 파일 변경 시각화

성공 지표: 첫 작업 완료까지 5분 이내

3.3 파일/폴더 접근

- ChatGPT와의 핵심 차별점
- 로컬 파일 시스템 전체 접근
- 실제 파일 생성/수정/삭제 가능

3.4 기본 프리셋 3개

프리셋 = 미리 설정된 AI 업무 환경

```
프리셋 구조:  
my-preset/  
|--- CLAUDE.md      # AI 지시사항 (역할, 톤, 규칙)  
|--- manifest.json   # 메타데이터, 필요 MCP  
|--- commands/       # 슬래시 커맨드  
└--- templates/      # 문서 템플릿
```

기본 제공 프리셋 (안):

- 마케팅 어시스턴트 - SNS 콘텐츠, 분석
- 웹 리서처 - 조사, 출처 정리, 보고서
- 문서 작성 도우미 - 기획서, 보고서, 회의록

3.5 자동 업데이트

플랫폼	방식
macOS	Squirrel (electron autoUpdater)
Windows	Inno Setup + 백그라운드 업데이트

4. 프리셋 vs Genspark 에이전트 비교

Genspark 에이전트의 실체

Genspark는 "Mixture-of-Agents" 아키텍처 사용:

- 8~9개 LLM 동시 사용 (GPT-4, Claude, Gemini, DeepSeek 등)
- 80개+ 자체 도구 내장
- Claude가 오케스트레이터 역할
- 자동 모델 라우팅: 작업 복잡도에 따라 최적 모델 선택
- \$275M 투자, \$1.25B 밸류에이션

Genspark 비즈니스 모델

플랜	가격	대상
Free	\$0	일반 사용자
Starter	\$19/월	프리랜서/블로거
Pro	\$79/월	콘텐츠 집중 팀
Team (B2B)	\$30/유저/월	기업

핵심: 사용자는 단일 요금만 지불, Genspark가 여러 AI 비용 부담

비교표

항목	Genspark 에이전트	Lili 프리셋
본질	멀티 LLM 오케스트레이션	설정값 패키지
AI 모델	8-9개 LLM 조합	Claude 단일

항목	Genspark 에이전트	Lili 프리셋
자체 도구	80개+ 내장	MCP 의존
오케스트레이션	자동 모델 라우팅	없음
파일 접근	제한적	로컬 전체
커스터마이징	제한적	사용자 수정 가능
기술 복잡도	매우 높음	상대적으로 단순

Lili 프리셋의 정확한 정의

"특정 업무를 위해 미리 작성된 CLAUDE.md + 슬래시 커맨드 + MCP 설정 묶음"

Genspark처럼 여러 AI를 오케스트레이션하는 게 아니라, Claude 하나에 좋은 프롬프트와 도구를 미리 연결해주는 것

Genspark의 경쟁력: 왜 무시할 수 없는가

압도적인 투자와 성장

지표	내용
Series B	\$275M (2025년 11월)
밸류에이션	\$1.25B (유니콘)
성장 속도	런칭 후 1년 내 유니콘 달성

기술적 해자 (Moat)

- Mixture-of-Agents:** 단일 모델이 아닌 여러 전문 모델의 조합으로 더 나은 결과
- 자동 라우팅:** 사용자가 모델을 선택할 필요 없이 최적 모델 자동 배정
- 80개+ 내장 도구:** MCP 연동 없이도 바로 사용 가능

비즈니스 모델의 강점

- 사용자는 단일 요금만 지불 (AI 비용 고민 없음)
- 복잡한 API 키 관리, 요금 계산 불필요
- 진입장벽 최소화 → 빠른 사용자 확보

왜 Lili는 Genspark와 다른 방향인가

관점	Genspark	Lili
핵심 가치	AI 성능 극대화	비개발자 접근성
복잡도	높음 (멀티 LLM)	낮음 (Claude 단일)
파일 접근	클라우드 중심	로컬 파일 전체
커스터마이징	제한적	프리셋/MCP로 확장
타겟	콘텐츠 파워유저	비개발자 일반 업무

결론: Genspark는 "AI 성능"으로 경쟁, Lili는 "접근성 + 로컬 파일"로 차별화

Genspark와 같은 방향으로 경쟁하면 자본/기술력에서 밀림. **다른 전장에서 싸워야 함.**

5. 비즈니스 모델 옵션

Claude API 상업적 사용

가능합니다. Claude API를 이용해 서비스를 만들고 단일 요금을 받는 것은 허용됨.

항목	내용
Output 소유권	사용자(팝업스튜디오)가 소유
재판매/서비스화	허용됨
저작권 보호	Anthropic이 저작권 침해 소송 방어
데이터 학습	Commercial Terms는 학습에 사용 안 함

Lili 비즈니스 모델 옵션

모델	설명	장점	단점
A. BYOK	사용자가 자기 API 키 사용	AI 비용 부담 없음	진입장벽 높음

모델	설명	장점	단점
B. 통합 구독	Lili가 API 비용 부담, 단일 요금	진입장벽 낮음	마진 관리 필요
C. 하이브리드	기본 무료 + 프리미엄 구독	유연함	복잡도 증가

6. 협업 기능 검토

6.1 왜 협업 기능을 고민하게 되었나?

문제 인식: "얇은 래퍼"의 한계

Lili Agent의 본질은 Claude Code CLI의 GUI 래퍼입니다. 이 "얇은 래퍼" 접근법은 빠른 개발이 가능하지만, 근본적인 문제가 있습니다:

위험	설명
Anthropic 공식 GUI 출시	Anthropic이 공식 Claude Desktop GUI를 출시하면 Lili의 존재 이유가 사라짐
낮은 전환 비용	사용자가 다른 도구로 쉽게 이동 가능
차별점 부재	단순히 "쉬운 설치"만으로는 지속적 경쟁 어려움

협업이 해결책이 될 수 있는 이유

관점	설명
개인용 → 팀용	개인 도구는 대체 가능, 팀 도구는 전환 비용 높음
데이터 Lock-in	팀 히스토리, 프리셋, 프롬프트가 Lili에만 저장되면 이탈 어려움
B2B 가치	팀 플랜은 개인 플랜보다 객단가 높음
네트워크 효과	한 명이 쓰면 팀 전체로 확산

고민의 출발점

"Anthropic이 공식 GUI를 출시해도 살아남으려면, 단순 래퍼 이상의 가치가 필요하다"

이 질문에 대한 답으로 "팀 협업"을 검토하게 됨.

6.2 협업 기능: 6가지 방향

방향 1: 결과물 중심 협업 (Output-Centric)

개념: AI가 만든 결과물을 공유하고 함께 다듬기

기능	설명
결과물 공유 링크	"이 보고서 공유" → URL 생성
코멘트/피드백	공유된 문서에 팀원 피드백
버전 관리	AI 수정 히스토리 추적
내보내기	Google Docs, Notion, Slack으로 전송

장점: 구현 단순, 기존 도구와 연동 / 단점: Lock-in 약함

방향 2: 맥락 중심 협업 (Context-Centric)

개념: 작업 맥락(대화, 설정)을 공유해서 팀원이 이어받기

기능	설명
맥락 스냅샷	"이 상태에서 이어하기" 버튼
팀 프리셋	팀 공통 CLAUDE.md, 브랜드 가이드
프롬프트 라이브러리	검증된 프롬프트 팀 공유
작업 템플릿	"주간 보고서 작성" 같은 반복 작업

장점: 노하우 축적, 온보딩 용이 / 단점: "맥락"의 가치가 명확하지 않을 수 있음

방향 3: 워크플로우 중심 협업 (Workflow-Centric)

개념: 여러 사람이 순차적으로 참여하는 업무 프로세스

기능	설명
작업 핸드오프	A가 초안 → B가 검토 → C가 승인
알림/멘션	"@김팀장 검토 부탁드립니다"
상태 관리	초안/검토중/승인/완료
승인 워크플로우	특정 결과물은 승인 필요

장점: 실제 업무 프로세스 반영 / **단점:** 개발 복잡도 높음, Jira/Asana와 경쟁

방향 4: 지식 중심 협업 (Knowledge-Centric)

개념: 팀이 AI와 함께 쌓아가는 지식 베이스

기능	설명
팀 메모리	AI가 팀 맥락을 기억 (회사 정보, 프로젝트 등)
자동 문서화	AI 작업 결과 자동으로 위키화
Q&A 축적	"이전에 누가 이거 물어봤었는데..."
검색	팀 전체 AI 작업 이력 검색

장점: 강력한 Lock-in, 시간이 갈수록 가치 증가 / **단점:** 초기에 가치 체감 어려움

방향 5: 실시간 협업 (Real-time)

개념: 같은 AI 세션에 여러 명이 동시 참여

기능	설명
공동 세션	여러 명이 같은 Claude와 대화
화면 공유	AI 작업 화면 실시간 공유

기능	설명
커서 표시	누가 어디서 작업 중인지
음성 채팅	작업하면서 대화

장점: 몰입감, 즉각적 협업 / 단점: 기술 복잡도 매우 높음

방향 6: 연동 중심 협업 (Integration-Centric)

개념: 기존 협업 도구와 깊게 연동

기능	설명
Slack 연동	Lili 결과물 → Slack 채널 자동 공유
Notion 연동	AI 결과물 → Notion 페이지 자동 생성
Google Drive	결과물 자동 저장
Jira 연동	작업 완료 → 티켓 상태 변경

장점: 기존 습관 유지, 진입장벽 낮음 / 단점: Lili 자체 Lock-in 약함

방향별 비교 매트릭스

방향	개발 복잡도	Lock-in	차별화	MVP 적합성
결과물 중심	낮음	약함	약함	★★★
맥락 중심	중간	중간	중간	★★★
워크플로우 중심	높음	강함	중간	★
지식 중심	높음	매우 강함	강함	★★
실시간	매우 높음	중간	강함	★
연동 중심	중간	약함	약함	★★

MVP 추천 조합: 맥락 중심 + 지식 중심

왜 "연동 중심" 대신 "지식 중심"인가?

비교 항목	연동 중심	지식 중심
Lock-in	약함 (Slack/Notion이 본체)	매우 강함 (데이터가 Steffy에 축적)
차별화	약함 (MCP로 누구나 가능)	강함 (팀 메모리는 쉽게 복제 불가)
장기 가치	정체	시간이 갈수록 증가
경쟁자 대비	Slack/Notion 봇과 동일	Claude Team에도 없는 기능

핵심 근거

1. 연동은 후순위로 미뤄도 됨

- Slack/Notion 내보내기는 MCP로 언제든 추가 가능
- MVP에 없어도 사용자 경험에 치명적이지 않음

2. 지식 축적이 진짜 해자(Moat)

- 팀이 3개월 쓰면 "떠나기 어려운" 상태가 됨
- 경쟁 제품으로 마이그레이션 비용 높음

3. "초기 가치 체감 어려움" 우려 해소

- 팀 프리셋/브랜드 가이드는 첫날부터 가치 있음
- "팀 메모리 축적"은 시간이 지나면서 부가 혜택으로 체감
- MVP에서는 간단 버전(대화 히스토리 저장 + 검색)만 구현

MVP 기능 목록 (수정)

- 팀 프리셋 - 팀 공통 CLAUDE.md/브랜드 가이드
- 프롬프트 라이브러리 - 검증된 프롬프트 공유
- 맥락 공유 링크 - "이 상태에서 이어하기"
- 팀 메모리 (간단 버전) - AI가 팀 맥락 기억 (회사 정보, 프로젝트 등)
- 대화 히스토리 검색 - 팀 전체 AI 작업 이력 검색

후순위 (Phase 2)

- Slack/Notion 내보내기 (MCP 연동)

- 자동 문서화 (AI 작업 결과 위키화)
- Q&A 축적 고도화

이 조합의 장점:

- **Lock-in 극대화**: 쓸수록 떠나기 어려움
- **차별화 강화**: Claude Team에도 없는 "팀 메모리"
- 실제 업무에서 즉시 체감 (프리셋/프롬프트는 첫날부터)
- 시간이 갈수록 팀 지식 자산 축적

6.3 협업 중심 경쟁 제품 분석

채팅 UI 기반 Claude/AI 래퍼 중 팀 협업에 초점을 맞춘 제품들을 조사했습니다.

Dust (dust.tt)

포지셔닝: "Enterprise AI Agent Platform"

항목	내용
핵심 컨셉	여러 AI 에이전트가 팀처럼 협업
Agent Teams	에이전트끼리 서로 호출하며 복잡한 작업 수행
Agent Memory	에이전트가 과거 대화/맥락 기억
연동	Slack, Notion, GitHub, Google Drive + MCP 지원
ARR	~\$36M (2025 추정)

대표 사례: Qonto (핀테크) - 연간 50,000시간 절약

Lili와의 차이점:

- Dust는 "에이전트 팀" 개념 (여러 AI 협업)
- Lili는 "사람 팀" 개념 (여러 사람이 AI와 협업)

TeamAI (tryteamai.com)

포지셔닝: "Team AI Workspace"

항목	내용
핵심 기능	AI 대화 공유, 팀 프롬프트 라이브러리
대화 공유	대화를 팀원과 공유/포크 가능
지식 관리	팀 지식베이스 구축
가격	~\$30/user/월

특징:

- ChatGPT/Claude 등 멀티모델 지원
- 대화 히스토리 검색 및 공유
- 권한 관리 (Admin/Member)

Lili와 비교:

- TeamAI는 순수 채팅 UI (파일 접근 없음)
- Lili는 로컬 파일 시스템 접근 가능
- TeamAI가 이미 "팀 프롬프트 라이브러리" 기능 보유

CoSpaceGPT (cospacegpt.com)

포지셔닝: "Secure Multi-Model AI Workspace"

항목	내용
핵심 기능	보안 중심 팀 AI 워크스페이스
Shared Projects	프로젝트별 대화/맥락 공유
멀티모델	GPT-4, Claude, Gemini 등 지원
가격	\$20-25/user/월

특징:

- SOC2/HIPAA 컴플라이언스
- SSO/SAML 지원
- 사용량 대시보드

타겟: 보안 민감 기업 (금융, 의료)

Claude Team/Enterprise (공식)

포지셔닝: Anthropic 공식 팀 플랜

항목	내용
Projects 기능	200K 컨텍스트 + 팀 공유 지식
협업	프로젝트별 역할 기반 권한
실시간	같은 프로젝트에서 실시간 협업
가격	\$25-30/user/월

핵심 기능:

- Project Instructions (팀 공통 지시사항)
- Shared Knowledge Base (파일 업로드)
- Activity Log (팀원 활동 추적)

경쟁 제품 비교 매트릭스

제품	파일 접근	팀 협업	멀티모델	가격	차별점
Dust	★★★ (연동)	★★★★★★	★★★★★	높음	에이전트 팀
TeamAI	×	★★★★★	★★★★★	중간	대화 공유/포크
CoSpaceGPT	★ (제한)	★★★★★	★★★★★	중간	보안/컴플라이언스
Claude Team	★★★	★★★★★	×	중간	공식, 안정성
Lili (목표)	★★★★★★	★★★ ~ ★★★★	×	TBD	로컬 파일 + MCP

경쟁 제품 분석에서 얻은 인사이트

인사이트	설명
파일 접근이 차별점	대부분 경쟁 제품은 클라우드 파일만 연동, 로컬 파일 직접 접근은 Lili만
프롬프트 라이브러리는 기본	TeamAI, Claude Team 모두 이미 보유 → Lili도 필수
대화 공유/포크는 쉽게 구현 가능	MVP에 포함 고려
Claude Team이 직접 경쟁자	하지만 로컬 파일 접근 없음이 약점
Dust는 다른 시장	에이전트 오케스트레이션은 다른 카테고리

7. 미결정 사항

질문 1: MVP에 팀 기능 포함 여부

옵션	장점	단점
A. 포함	초기부터 차별화, Lock-in 구축 시작	개발 복잡도 증가, 출시 지연
B. 제외	빠른 출시, 검증 후 결정	Anthropic GUI 출시 시 차별점 없음

질문 2: 장기 차별화 전략

옵션	설명
A. 팀 협업 (넓이)	Notion AI, Slack과 경쟁
B. 직군 전문화 (깊이)	마케터 or 리서처 전용 도구로 포지셔닝
C. 두 가지 병행	MVP는 개인용, 이후 사용자 피드백 보고 방향 결정

질문 3: 비즈니스 모델

옵션	설명
A. BYOK	사용자가 Claude API 키 직접 입력
B. 통합 구독	Lili가 AI 비용 부담, 단일 요금
C. 하이브리드	무료 (BYOK) + 프리미엄 (통합)

8. 기술 스택 (참고)

영역	기술
프레임워크	Electron
프론트엔드	React + TypeScript
UI	Tailwind CSS + shadcn/ui
터미널	xterm.js + node-pty
상태 관리	Zustand
빌드	electron-builder
токен 보안	keytar (OS Keychain)
백엔드 (팀 기능 시)	bKend (자체 BaaS)

9. 다음 단계

- 팀 논의: 위 결정 사항에 대해 김경호님/팀과 논의
- 방향 결정: 팀 협업 vs 직군 전문화 vs 병행
- 비즈니스 모델 결정: BYOK vs 통합 구독 vs 하이브리드
- 사용자 검증: 초기 사용자 인터뷰로 팀 기능 필요성 검증
- 상세 요구사항: 결정 후 기능별 상세 스펙 정의

참고 자료

Genspark

- [VentureBeat - Genspark Architecture](#)
- [Genspark \\$275M Series B](#)
- [Sacra - Genspark Revenue & Valuation](#)

Claude 사용 사례

- [Anthropic - How Anthropic Teams Use Claude Code](#)
- [Anthropic - Enterprise AI Transformation](#)
- [Claude Customer Stories](#)
- [Product Talk - Claude Code for Non-Technical People](#)
- [Animalz - Claude Code for Content Marketers](#)

Claude API 정책

- [Anthropic - Expanded Legal Protections](#)
- [Terms.law - Claude Output Ownership](#)