

시작

서론

Part 1

Part 2

Part 3

결론

바이브 기획

AI 시대, 기획자의 새로운 무기

2024.12.06

01

시작
서론
Part 1
Part 2
Part 3
결론

기획의 생명은 속도

바이브 코딩 시대입니다.

바이브 기획, 바이브 MVP가 요구되고 있어요.

실제로 실리콘밸리에서는 "**바이브 PM**" 직무가 등장하고 있습니다.

오늘 주제

새로운 서비스를 만드는 상황을 전제로,

"바이브 기획"이란 무엇인지 함께 알아보겠습니다.

기획자의 현실

주변 기획자들에게 물어보면:

- 자료조사는 **Perplexity**
- 주로 **GPT** 사용
- UX 설계는 **Figma Make, Gemini**

문제는?

기획자를 위한 특출난 AI 서비스가 없습니다.

왜일까요?

기획은 점진적으로 구체화되는 작업이기 때문입니다.

자료조사 하나만 해도:

경쟁사 분석

SWOT 분석

포지셔닝 설정

시장 규모 파악

타겟 고객 리서치

벤치마킹

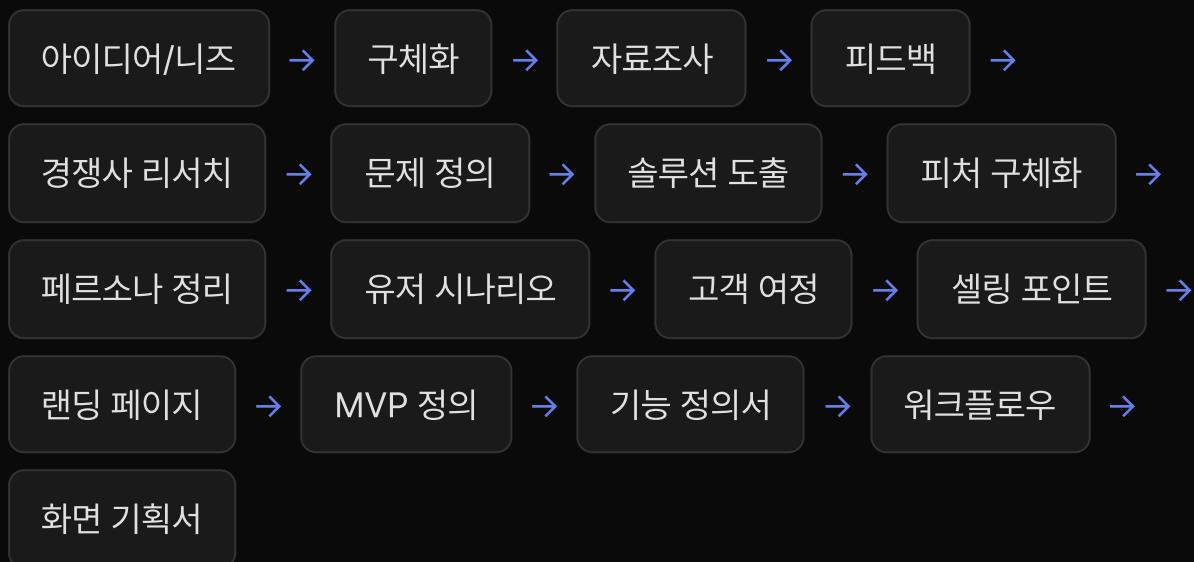
이 각각이 또 세부 태스크로 쪼개지고,
결과가 서로 영향을 주고,
피드백 받으면 다시 돌아가서 수정하고...

시작
서론
Part 1
Part 2
Part 3
결론

그래서 "버튼 하나로 기획 완료!" 같은 서비스가 나오기 어렵습니다.

기획자의 업무 범위

신규 서비스 기준



전통적 방식

오늘 목표

시작

서론

Part 1

Part 2

Part 3

결론

중요한 전제

"모든 것을 AI화 할 수 없습니다.

AI는 생각의 출발점을 제공해주고,

그 뒤에 휴먼 터치는 항상 필수입니다."

Part 1: 왜 Claude Code인가? 02

AI 서비스 피로감

자료조사

Perplexity

코드

Cursor, Copilot

UX

Figma AI, Figjam

글쓰기

ChatGPT, Notion AI

분석

Claude...

시작

서론

Part 1

Part 2

Part 3

결론

문제점

- 서비스마다 다른 UI 학습
- 컨텍스트가 분산됨
- 워크플로우 단절
- 결과물 통합의 어려움

"하나만 잘 쓰면 된다"

굳이 많은 AI 서비스를
다 알 필요 없습니다.

Claude Code가 좋은 이유:

1 컨텍스트 유지

한 세션에서 모든 작업

2 파일 시스템 접근

실제 문서 생성/수정

3 확장 가능

MCP, Custom Commands로 나만의 워크플로우 구축

4 대화형

Claude Code vs 다른 AI 도구

구분	ChatGPT/Claude Web	Cursor/Copilot	Claude Code
주 용도	대화/질문	코드 작성	모든 작업
파일 시스템	X	코드 파일만	모든 파일
워크플로우	수동	코드 특화	커스텀 가능
컨텍스트	대화 내	코드베이스	프로젝트 전체
기획 적합성	보조 도구	부적합	최적

핵심: Custom Commands

Claude Code의 진짜 힘 = **나만의 명령어**

/appkit.new "테니스 코트 예약 앱"

→ 서비스 컨셉, 타겟 고객, 핵심 가치 자동 도출

/appkit.spec 003-booking "메인화면에서 검색 후 바로 예약"

→ 기능 구체화, 화면 흐름, 비즈니스 룰 자동 작성

/appkit.customer

→ 페르소나, 고객 여정 자동 생성

포인트: AI를 "잘 쓰는" 게 아니라

"내 업무에 맞게" 만드는 것

03

Part 2: AppKit 프레임워크

시작
서론
Part 1
Part 2
Part 3
결론

아이디어를 고객 가치로 변환하는
논리적 사고 프레임워크

핵심 철학:

고객 가치 우선

기능이 아닌 고객의 문제 해결 중심

논리적 사고 연습

큰 작업을 작은 단계로 쪼개기

스토리텔링 기반

고객의 하루와 고민을 이해하고 설득

MVP 중심 실행

최소한의 범위로 최대한의 검증

논리적 사고 7단계

1. **/appkit.new** → 아이디어 스케치 (어떤 서비스인지?)
2. **/appkit.spec** → 기능 구체화 (뭐가 필요할까? 누가 쓸까?)
3. **/appkit.customer** → 고객 스토리 (고객의 하루, 고민, 해결)
4. **/appkit.sales** → 세일즈 랜딩 (어떻게 설득할까?)
5. **/appkit.mvp** → MVP 범위 (최소한으로 검증하려면?)
6. **/appkit.merge** → 기획 정돈 (흩어진 기획 통합)
7. **/appkit.design** → 개발 준비 (API, ERD, 기술 스펙)

데모: 테니스 코트 예약 앱

Step 1: 아이디어 스케치

/appkit.new Tennis court booking app with coupons and time deals

결과물:

- **서비스 컨셉:** "직장인의 주말 운동을 쉽게 만드는 예약 서비스"
- **핵심 문제:** 전화 예약 번거로움, 가격 불투명, 시간 확보 어려움
- **해결책:** 3초 예약, 가격 비교, 자동 할인

시작

서론

Part 1

Part 2

Part 3

결론

Step 2: 기능 구체화

/appkit.spec 003-booking "퇴근길 지하철에서 3초 예약"

결과물:

- 고객 가치 분석 (누가/언제/왜/어떻게)
- 화면 흐름 자동 생성
- 비즈니스 룰 도출

Step 3: 고객 스토리

/appkit.customer

김대리 (35세, IT기업)의 하루:

- 07:00 - "오늘도 운동 못하겠네...." (죄책감)
- 12:00 - "주말엔 테니스 쳐야지!" (결심)
- 18:00 - "코트 예약 전화... 귀찮다" (좌절)
- 19:00 - [앱 발견] "3초 만에 예약?" (희망)
- 19:02 - "토요일 오전 예약 완료!" (만족)

시작

서론

Part 1

Part 2

Part 3

결론

생성되는 산출물

```
docs/appkit/
└── overview.md # 서비스 컨셉
└── specs/
    ├── 001-search/spec.md # 기능별 상세
    ├── 002-booking/spec.md
    └── ...
└── customer-persona.md # 페르소나
└── customer-journey.md # 고객 여정
└── sales-landing.md # 세일즈 메시지
└── mvp-scope.md # MVP 범위
└── merge/ # 통합 문서
└── design/ # 개발 스펙
```

...이 모든 게 **하루 안에** 가능

Part 3: 여러분의 것을 만드세요

중요

AppKit 자체가 정답이 아닙니다.

AppKit은 하나의 예시입니다:

- 제가 기획 업무를 하면서 필요했던 것
- 제 사고방식을 반영한 것
- 제 업무 흐름에 맞춘 것

여러분의 업무는 다릅니다.

AppKit 오픈소스 공개

GitHub에서 확인하세요

github.com/rupy1014/appkit

- 프롬프트 구조와 명령어 설계를 참고하세요
- Fork해서 여러분 업무에 맞게 수정해서 쓰세요
- 정답을 가져가는 게 아니라, 만드는 방법을 가져가세요

시작

서론

Part 1

Part 2

Part 3

결론

핵심은 "나만의 프레임워크"

생각해볼 질문:

- 1 내 업무에서 반복되는 패턴은?
- 2 매번 같은 질문을 AI에게 하나?
- 3 자주 만드는 문서 형식이 있나?
- 4 팀에서 공유하면 좋을 워크플로우는?

Custom Command 만들기

Claude Code에서:

```
----  
description: 내 업무를 자동화하는 명령어  
----  
## 입력  
$ARGUMENTS  
  
## 실행할 작업  
1. [여러분의 업무 단계]  
2. [여러분의 체크리스트]  
3. [여러분의 출력 형식]
```

시작
서론
Part 1
Part 2
Part 3
결론

시작하는 방법

1 관찰

일주일간 반복 업무 기록

2 패턴화

공통점 찾기

3 명령어화

Custom Command 작성

4 개선

사용하면서 다듬기

05

시작
서론
Part 1
Part 2
Part 3
결론

오늘의 핵심 메시지

AI 서비스 피로

다 알 필요 없다,
하나를 깊게

Claude Code

기획에 최적화된 도구

나만의 프레임워크

AI를 "쓰는" 게 아니라
"만드는" 것

시작

서론

Part 1

Part 2

Part 3

결론

"기획의 속도는 도구가 아니라
사고방식에서 나옵니다.

AI는 그 사고를 구체화하는 도구일 뿐,
여러분만의 방식을 만들어가세요."

시작
서론
Part 1
Part 2
Part 3
결론

참고 자료

Claude Code

claude.ai/code

AppKit 오픈소스

github.com/rupy1014/appkit

프롬프트와 명령어 구조 참고

Fork해서 여러분 업무에 맞게 수정해서 사용하세요

감사합니다 🌟