# 1. 프로젝트 Flow와 기획 단계



프로젝트가 전반적으로 흘러가는 flow

워터풀 방식, 애자일 방식의 차이

기획 단계에서는 어떤 것을 애기해야하는지

최소한의 MVP는?

## 프로젝트는 전반적으로 어떻게 흘러갈까?

프로젝트는 한 번 정하고 끝나는 일이 아닙니다.

대부분 아래와 같은 흐름으로 반복되며 진행됩니다.

```
아이디어
 → 기획
 → 개발
 → 배포
 → 운영
 → 회고
 → (다시 기획)
```

여기서 중요한 점은, 프로젝트는 직선이 아니라 순환 구조라는 점

~~운영까지 하면 무한 도돌이표입니다 흑흑~~

기획이 끝났다고 기획을 다시 안 하는 것도 아니고,

개발 중이라고 해서 기획을 못 바꾸는 것도 아닙니다.

다만, **각 단계마다 중심이 되는 일이 다를 뿐**입니다.

---

### 각 단계에서의 핵심 포인트

- **기획 단계**
  - 무엇을 만들 것인가
  - 무엇을 만들지 않을 것인가
- **개발 단계**
  - 어떻게 구현할 것인가
  - 지금 수준에서 충분한가
- **배포 / 운영 단계**
  - 실제로 사용 가능한가
  - 문제가 생기면 추적 가능한가
- **회고**
  - 잘한 결정은 무엇이었는가
  - 잘못된 결정은 왜 나왔는가

프로젝트가 힘들어지는 순간은 보통, 지금 우리가 어느 단계에 있는지 팀원마다 다르게 생각할 때입니다.
그래서 프로젝트 초반에 지금은 기획 단계인지, 개발 단계인지 명확히 인지하고 가는 게 중요하다 생각합니닷

## 워터폴 방식과 애자일 방식의 차이

개발 프로젝트 얘기를 하다 보면 꼭 나오는 말이 있습니다. 하핫 저도 언급했지만
“이거 애자일로 하게 가면 좋을거같아요”
그런데 정작 애자일이 뭔지 정확히 설명하기는 어렵습니다... 진짭니다

---

### 워터폴 방식

워터폴은 말 그대로 **폭포수처럼 한 단계씩 내려오는 방식**

1. 기획
2. 설계
3. 개발
4. 테스트
5. 배포

장점

- 단계가 명확하다
- 계획이 잘 서 있으면 안정적이다

단점

- 한 번 정한 기획을 바꾸기 어렵다
- 초반 실수가 끝까지 영향을 준다

### 애자일 방식

애자일은 **짧은 주기로 반복하면서 개선하는 방식**

- 완벽한 계획보다는 빠른 실행
- 작은 단위로 만들고 검증
- 변화에 빠르게 대응

장점

- 변경에 유연하다
- 실제 사용성을 빨리 확인할 수 있다

단점

- 기준이 없으면 방향이 흔들린다
- 경험이 없으면 더 혼란스러울 수 있다

---

여러 프로젝트를 진행하면서 수없이 기획이 엎어지기도 했고, 변경되기도 했습니다

첫 실수가 끝까지 가져간다는걸 알기에 더 확실하게 기획하기 위해 맹목적으로 생각했던 적도 있었고, 그렇기에 확장성 있게 생각하고 짧은 주기로 가져가는게 기획에 맹목적이지 않아 시간을 여유롭게 가져갈 수 있더라구요오

단, 애자일이 만능도 아니고 워터풀이 잘못된건 아닙니다

프로젝트의 성격에 맞게 방법론을 채택하는 거죠

(방법론은 v형, 프로토타입 등등 여러가지 많아요! 소프트웨어 공학하면 다루게 됩니다)

## 기획 단계에서는 무엇을 이야기해야 할까?

기획 단계라고 하면 많은 사람들이 이렇게 생각합니다.

기획자는 문서 만드는 사람 아닌가요?

하지만 기획의 본질은 문서가 아닙니다.

기획은 결정의 범위를 정하는 단계라고 생각합니다

### 기획 단계에서 반드시 나와야 할 질문들

- 이 서비스는 누구를 위한 것인가?
- 이 문제를 꼭 지금 풀어야 하는 이유는?

- 이 기능이 없으면 서비스가 성립하지 않는가?
- 나중에 해도 되는 것은 무엇인가?

이 질문에 답하지 않은 채 개발을 시작하면

중간에 반드시 이런 말이 나옵니다

이건 기획 때 정했어야 했는데... ㅜ

### 기능 기획 vs 서비스 기획

- 기능 기획

  → 무엇을 만들 것인가
- 서비스 기획

  → **왜 이걸 만들어야 하는가**

프로젝트에서 가장 많이 놓치는 건 기능은 많은데 이유가 없는 상태입니다.

## 최소한의 MVP란 무엇일까?

MVP는 가장 빨리 검증할 수 있는 형태입니다.

### MVP를 정할 때 자주 하는 실수

- 기능을 너무 많이 넣는다
- “이것도 있으면 좋지 않을까?”가 반복된다
- 결국 아무 것도 완성되지 않는다

### MVP를 정하는 기준

아래 질문에 답해보면 도움이 됩니다.

- 이 기능이 없으면 사용자가 아무것도 못 하는가?
- 이 기능으로 우리가 가설을 검증할 수 있는가?
- 지금 만들지 않으면 안 되는가?

MVP는 **가장 작은 기능 묶음**이 아니라 가장 빨리 의미 있는 피드백을 받을 수 있는 상태로써 핵심적인 기능들을 말하는 겁니댜