

프론트 엔드 개발자가 알아야 하는 컴퓨터 공학 지식

소프트웨어 공학(SW Engineering)

소프트웨어 공학 | 프론트 엔드 개발자가 알아야 하는 CS 지식

강사 나동빈

프론트 엔드 개발자가 알아야 하는 컴퓨터 공학 지식

소프트웨어 공학(SW Engineering)

애자일(Agile)

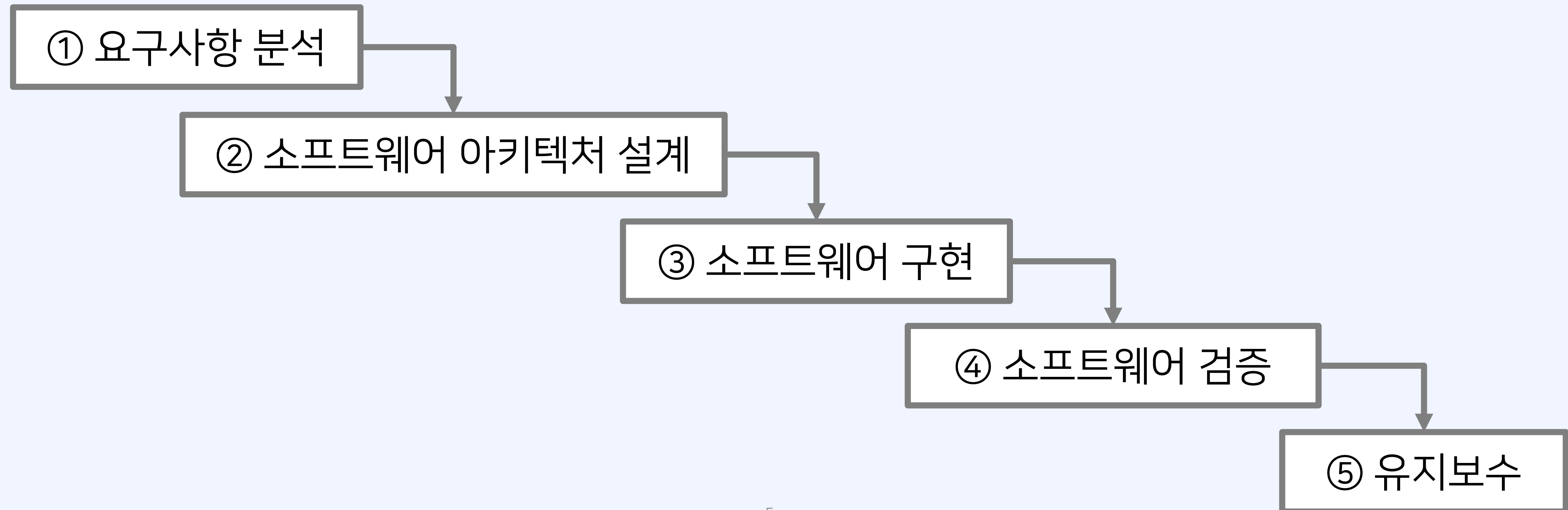
- 애자일은 소프트웨어 개발 방법론 중 하나이다.
 - 애자일은 사용자로부터 요구 사항을 모두 한꺼번에 받아서, 개발을 시작하지 않는다.
 - 애자일은 **일정한 주기**를 가지고 반복적으로 프로토타입을 만든다.
- 그때마다 필요한 요구 사항을 받아 수정하며 확장시킨다.

소프트웨어 개발 과정의 어려운 점

- 처음에 사용자로부터 받았던 개발 요구 사항은 계속해서 변경될 수 있다.
 - 개발 인원, 일정, 환경도 계속하여 변경될 수 있다.
- 따라서, 미리 정해진 계획만을 따르기 보다는 **변화에 대응할 수 있는 애자일 방식**이 채택되었다.

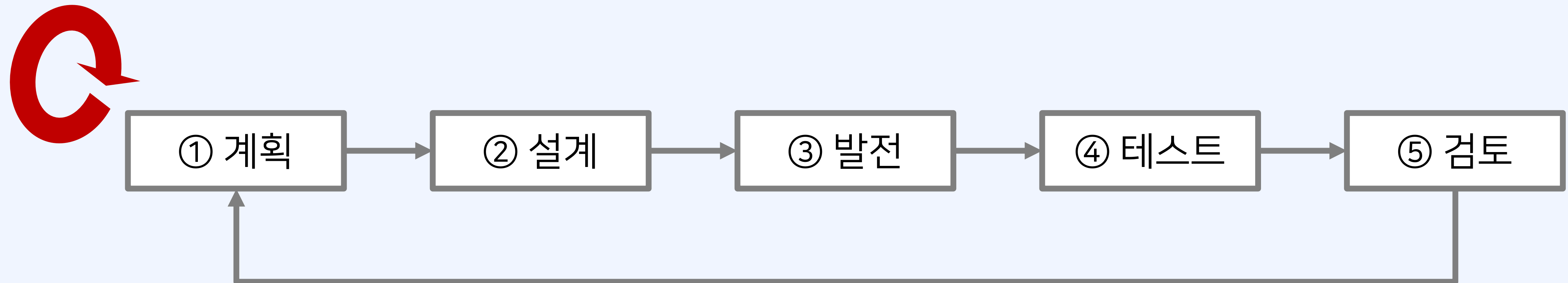
폭포수 개발 방식

- 가장 먼저, 사용자와 요구 사항 명세서에 대하여 합의를 진행한다.
- 이후에 아래와 같은 개발 과정을 순차적으로 진행한다.
- 이전 단계가 완료되기 전에는 다음 단계로 넘어갈 수 없다.



애자일(Agile)

- 애자일은 일정한 주기를 가지고 반복적으로 프로토타입을 만든다.
→ 그때마다 필요한 요구 사항을 받아 수정하며 확장시킨다.



애자일(Agile) 과정 예시

1. 프로그램에 대하여 필요한 기능을 명시한다.
2. 팀원들과 함께 스프린트(sprint)를 계획한다.
 - 스프린트: 소규모 기능에 대하여 계획, 개발, 테스트 과정을 포함하는 주기를 의미한다.
 - 회의에서 백 로그(스프린트를 위해 해야 할 일 목록)와 스프린트 목표를 다룬다.
3. 스크럼(scrum): 개발 과정에 대한 전체 주기(한 달 정도)
 - 일일 스크럼 회의를 거쳐, 날마다 간단히 한 일 및 직면한 문제 사항 등을 공유한다.
4. 스프린트 완료 이후에는 회의 및 검토를 거쳐 회고를 진행한다.

애자일(Agile) 적용하기

- Jira와 같은 프로그램을 사용하여 애자일 개발 방법론을 적용할 수 있다.
- 예를 들어, 매 **스프린트**에 대하여 다음과 같은 일들을 명시하여 작업할 수 있다.
 1. 해야 할 일(To Do)
 2. 현재 진행 중인 작업(In Progress)
 3. 완료된 작업(Done)