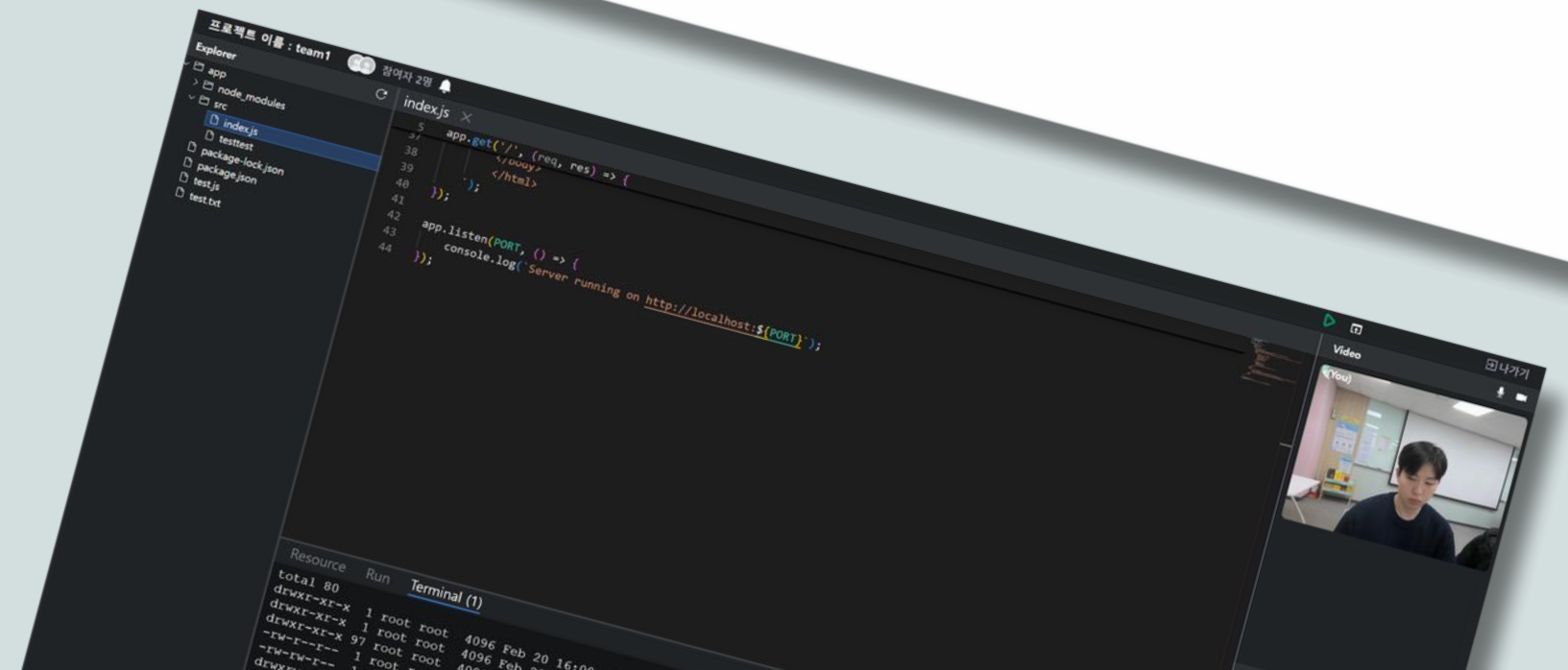
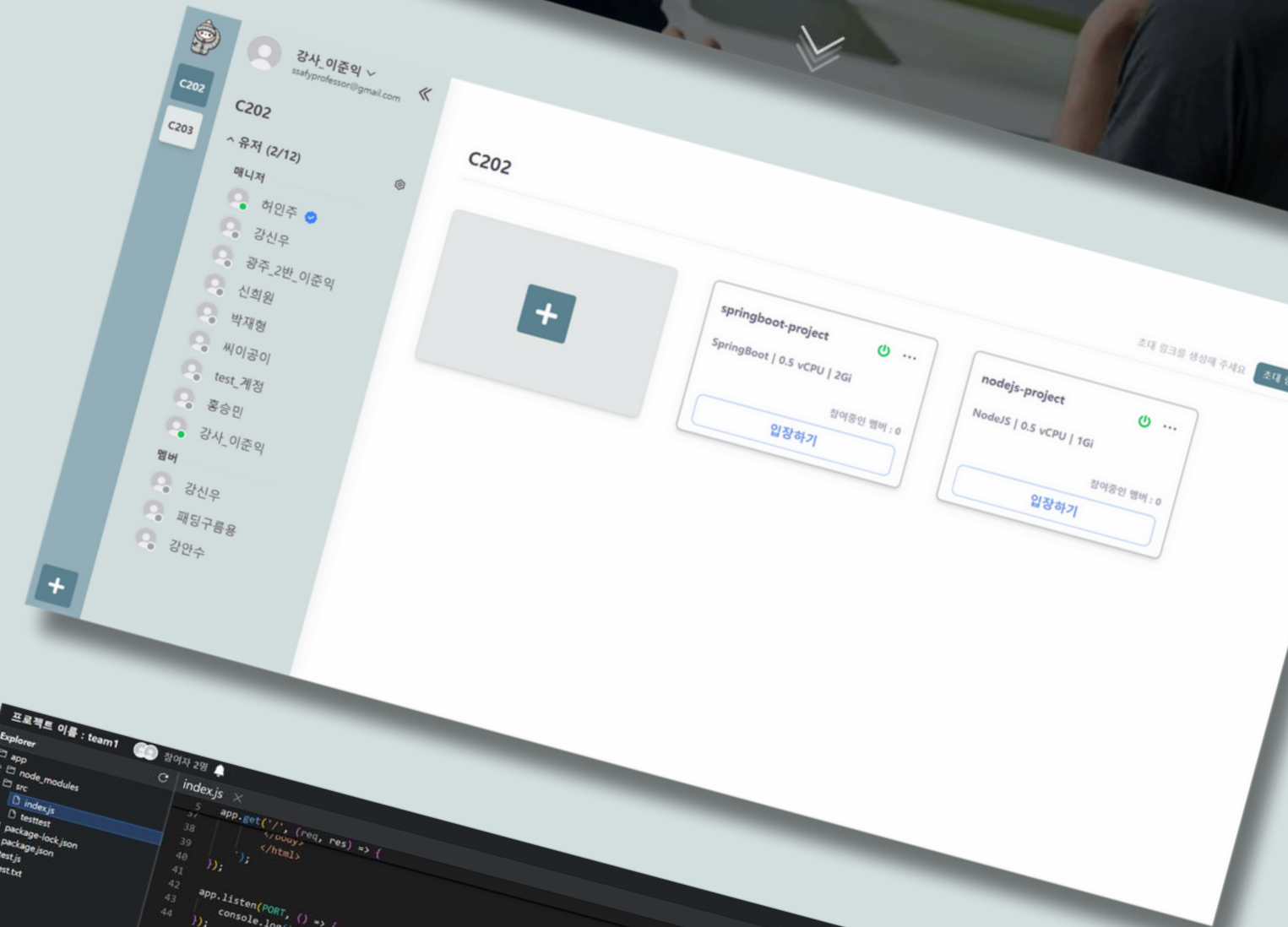


SSAFY 공통 프로젝트 C202팀



# PADING

페어 프로그래밍을 위한 웹 IDE 및 관리 시스템



# Contents

01 페어 프로그래밍

02 기획 배경

03 기능

04 시연

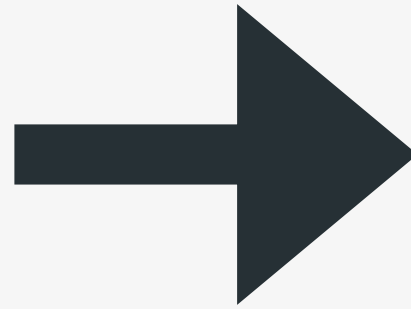
05 주요 기술 구현

06 실사용 후기

## 01 페어 프로그래밍이란?



Driver

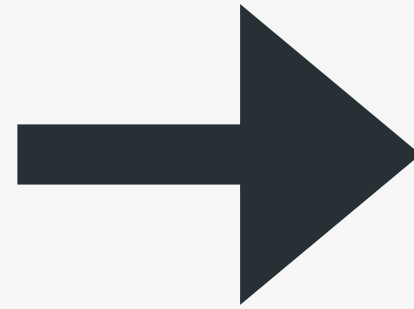
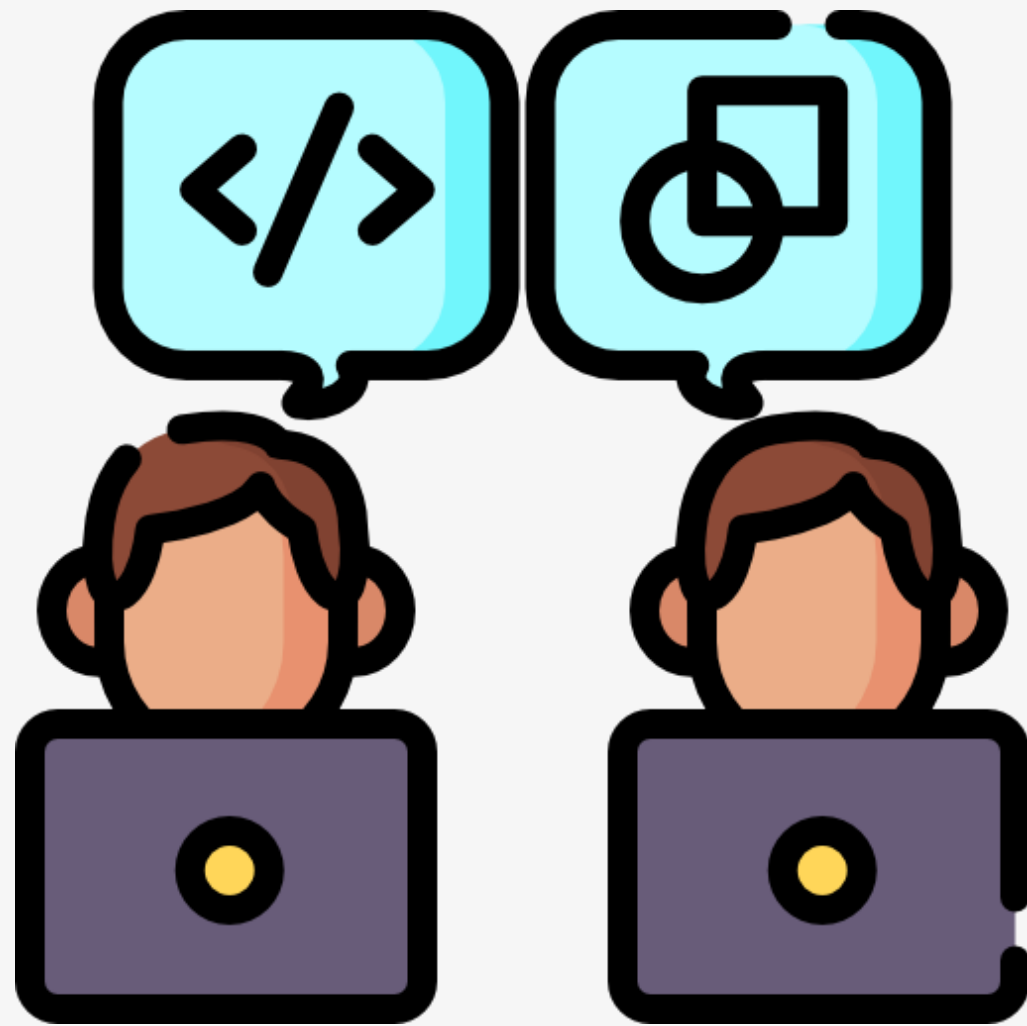


Observer

## 01 페어 프로그래밍의 장점



## 01 페어 프로그래밍의 장점



코드 품질 ↑

협업 능력 ↑

## 01 페어 프로그래밍 사례

 Inflearn

kakao

요기요





## 02 문제 정의



## 02 문제 정의

화상 채팅

동시 편집

매니징 시스템

자동 배포

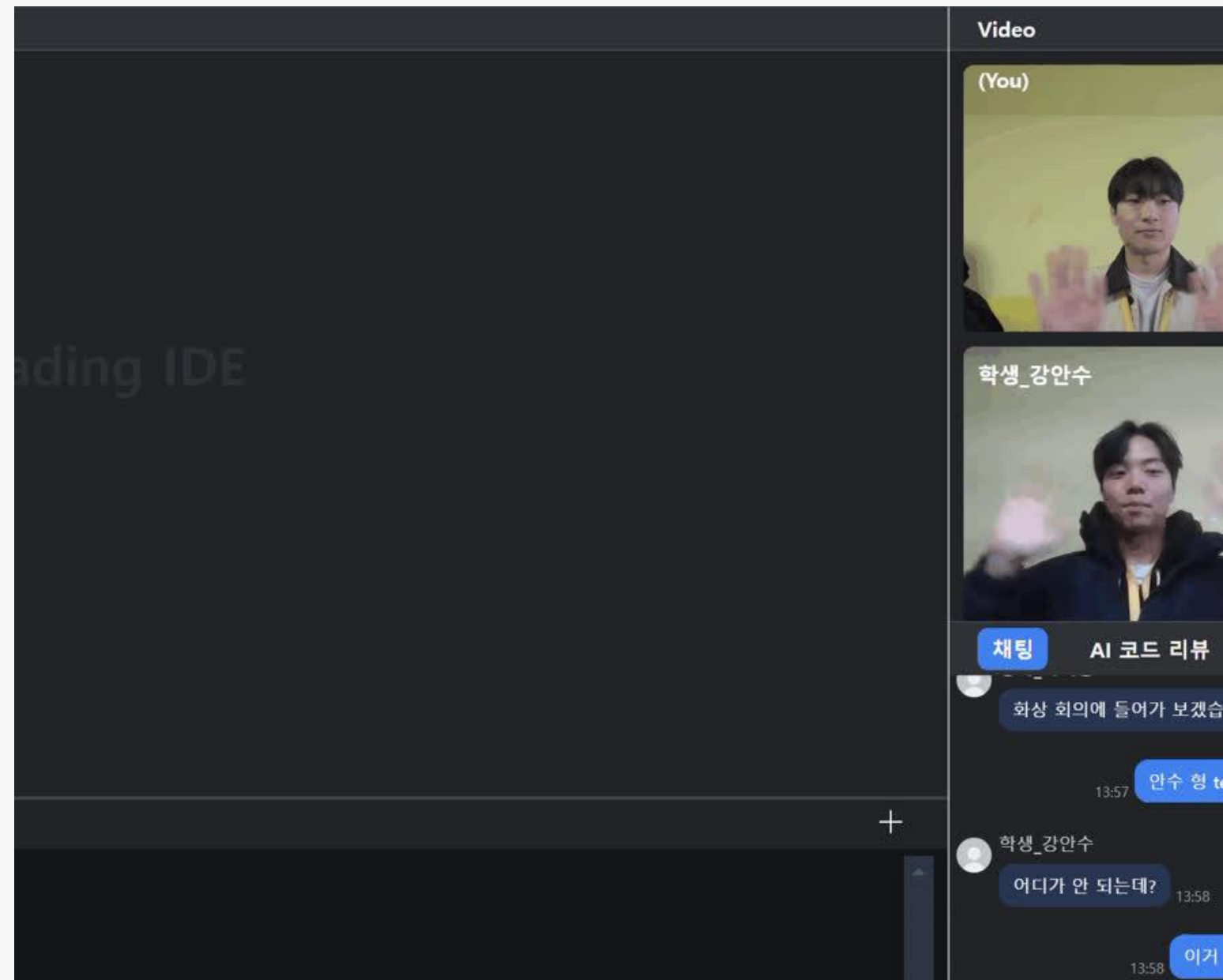




# PADING

Pair + Coding

## 화상 채팅



## 동시 편집

```
// 변수 선언 및 할당
let message = "Hello, world!";
let count = 0;

// 함수 정의
function greet(name) {
  return `Hello, ${name}!`;
}

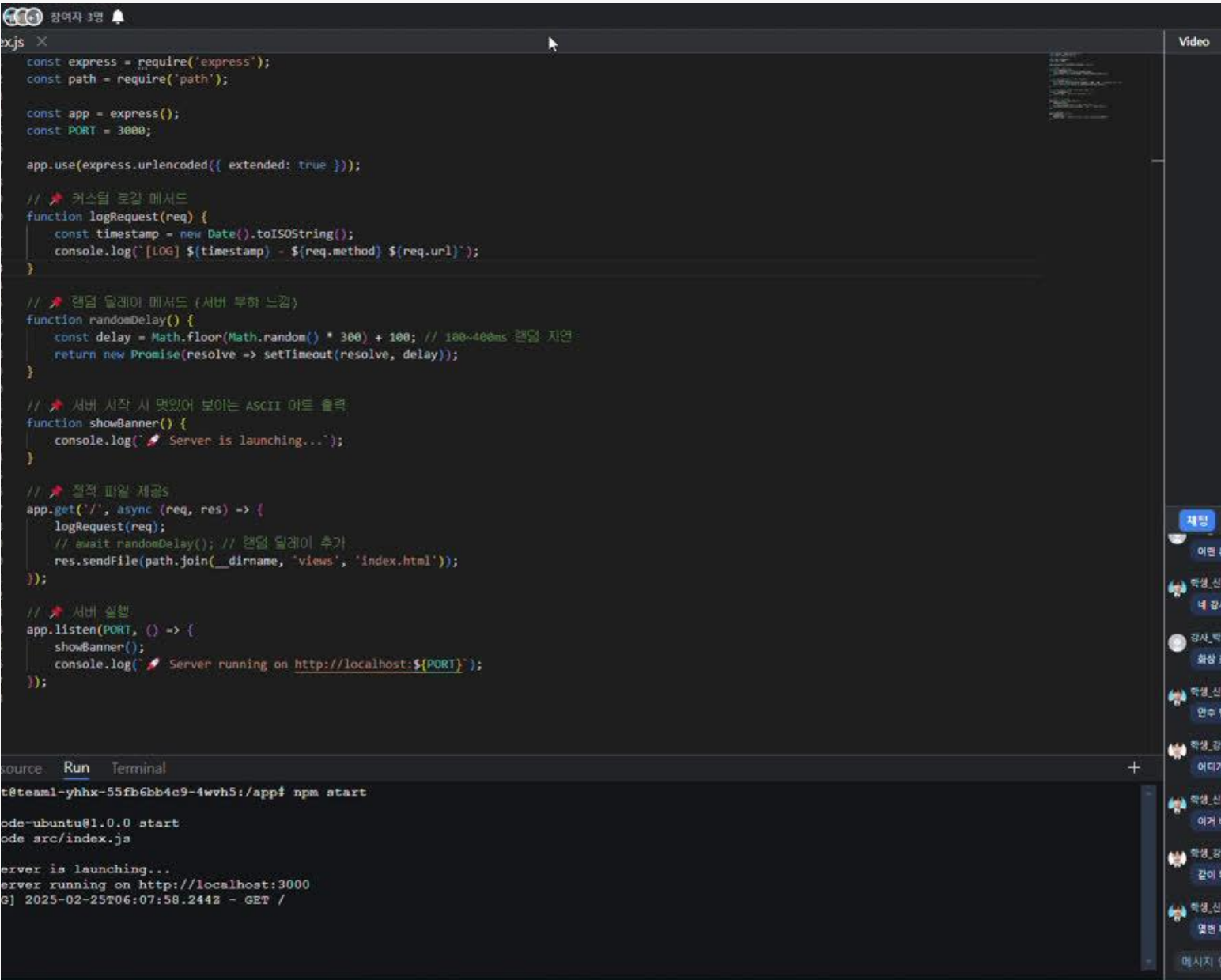
// 반복문 테스트
for (let i = 0; i < 5; i++) {
  console.log(greet("User"));
  count++;
}

// 객체 및 배열 활용
const user = {
  name: "John Doe",
  age:
};

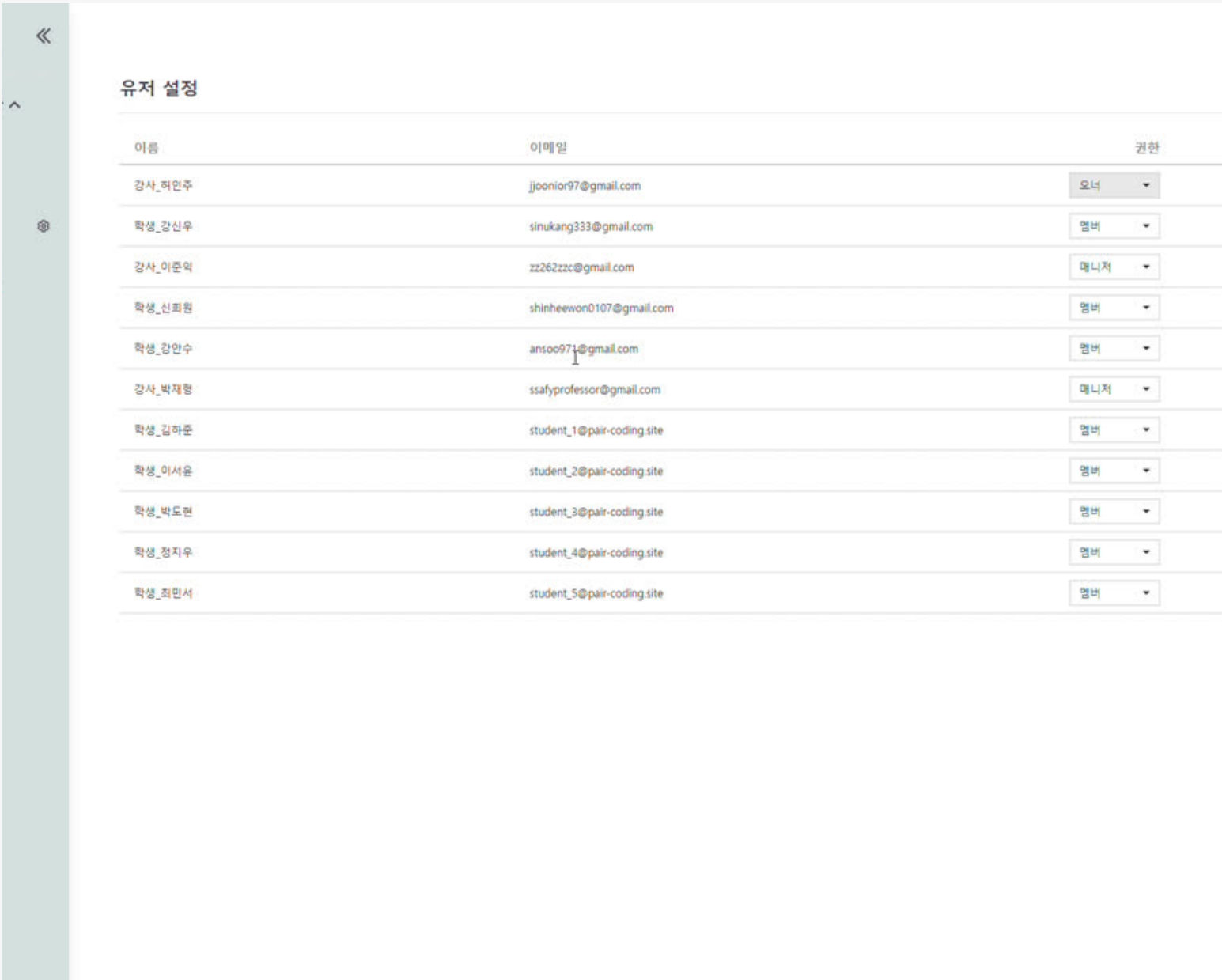
console.log(`User: ${user.name}, Age: ${user.age}`);
console.log("Hobbies:", user.hobbies.join(", "));

// 비동기 함수 테스트
async function fetchData() {
  return new Promise(resolve => {
    setTimeout(() => {
      resolve("Data loaded successfully!")
    });
  });
}
```

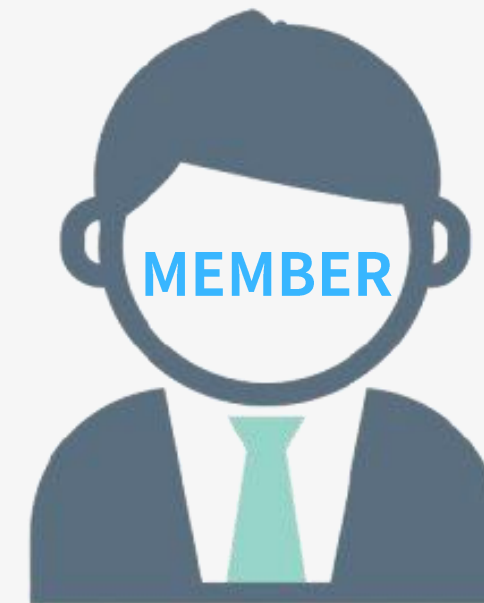
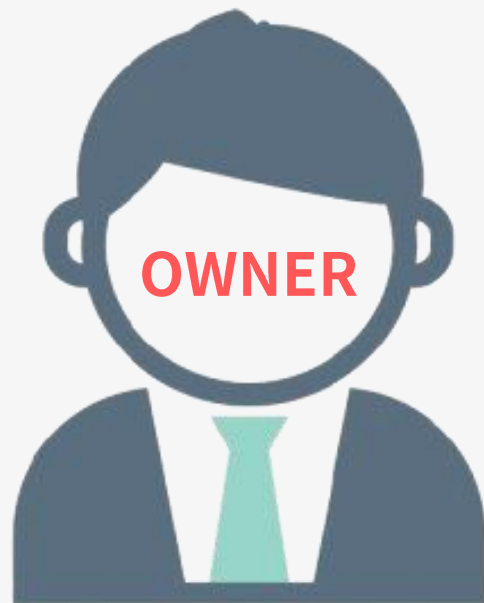
# 자동 배포



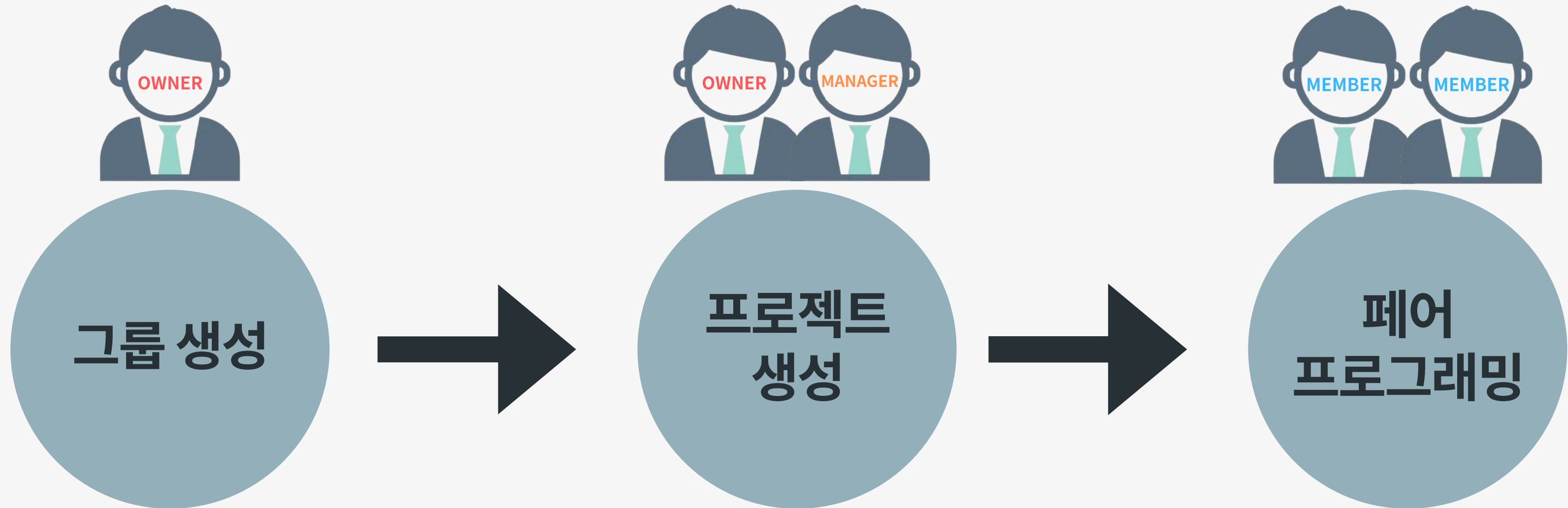
# 매니징 시스템



## 03 매니징 시스템



## 03 매니징 시스템



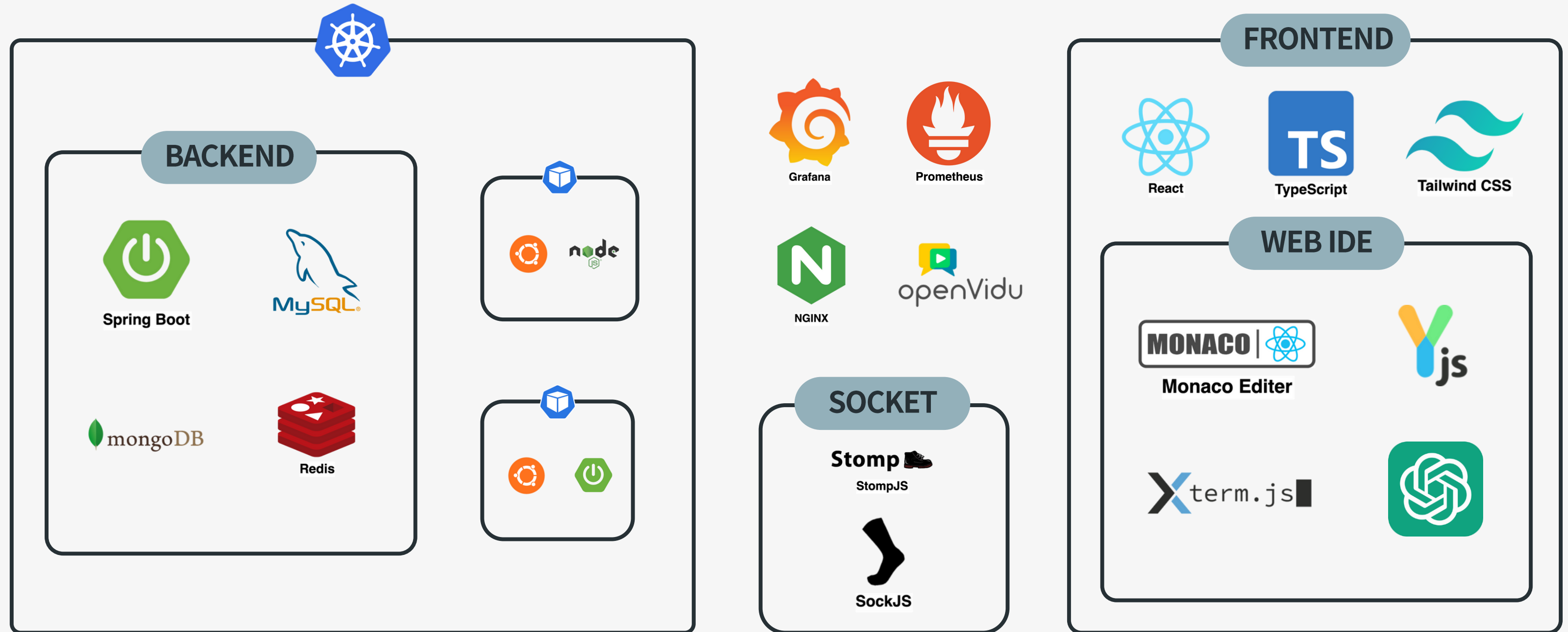
## 04 시연



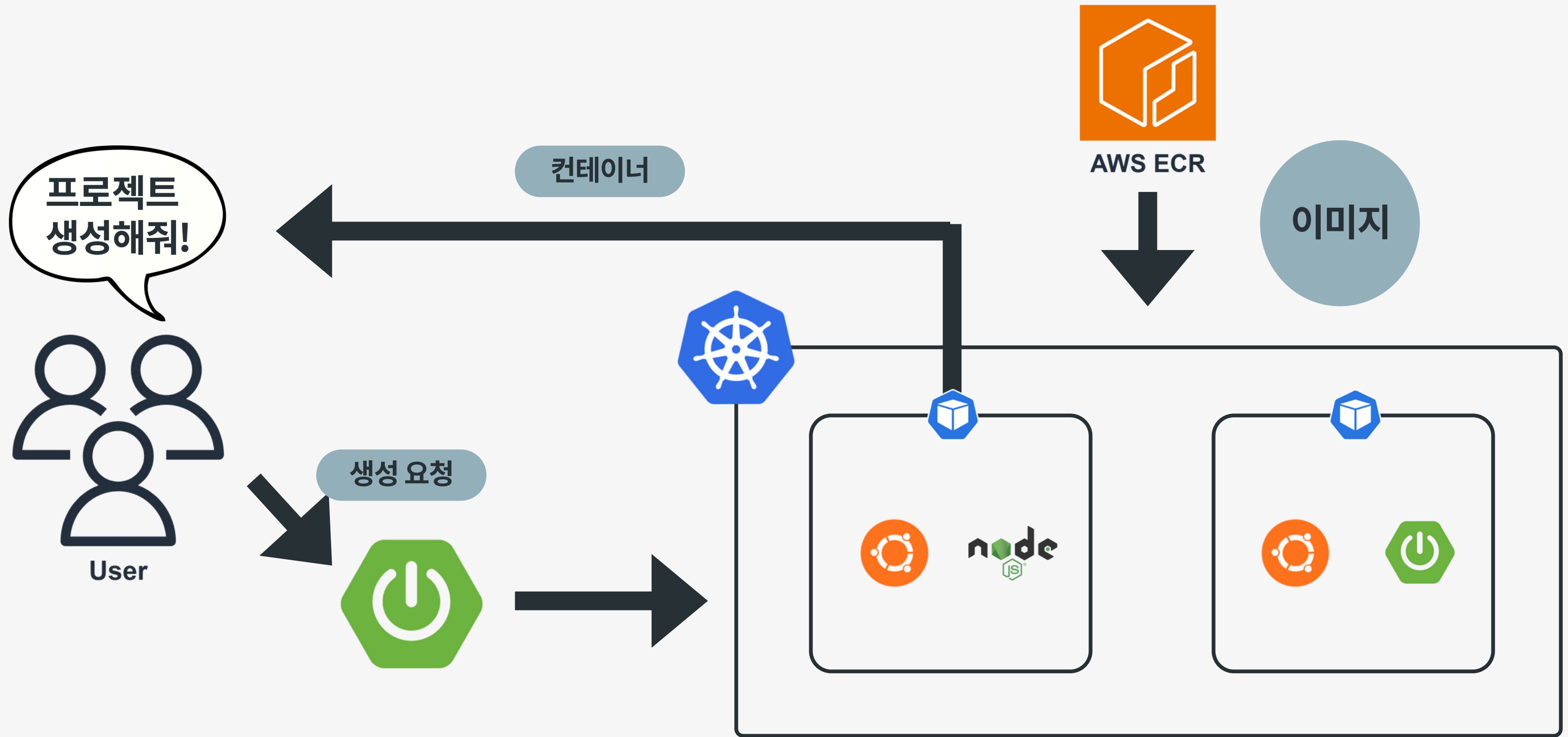
**LIVE 시연**

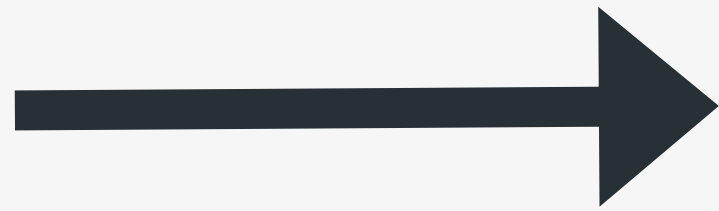


## 05 전체 아키텍처

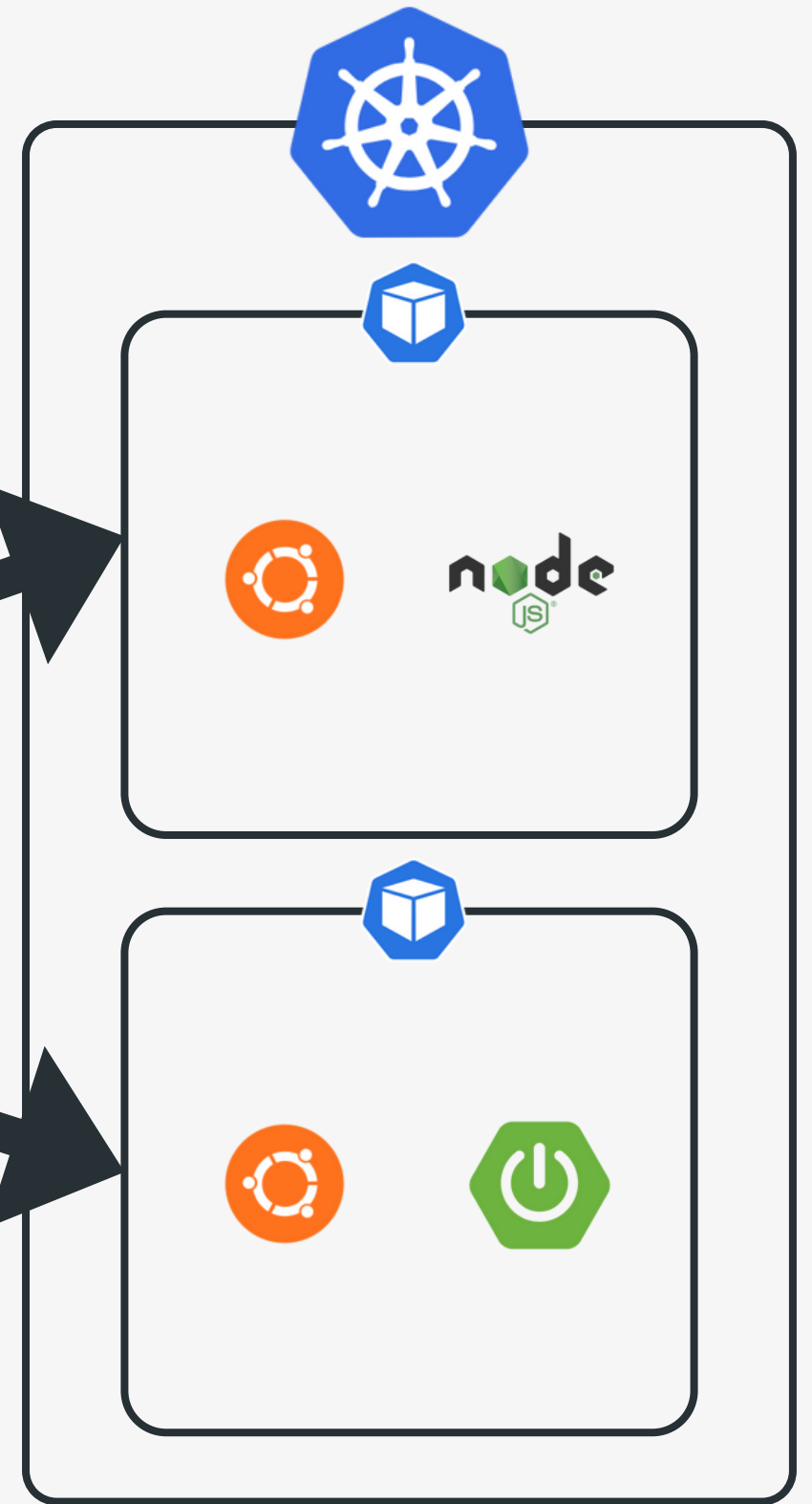
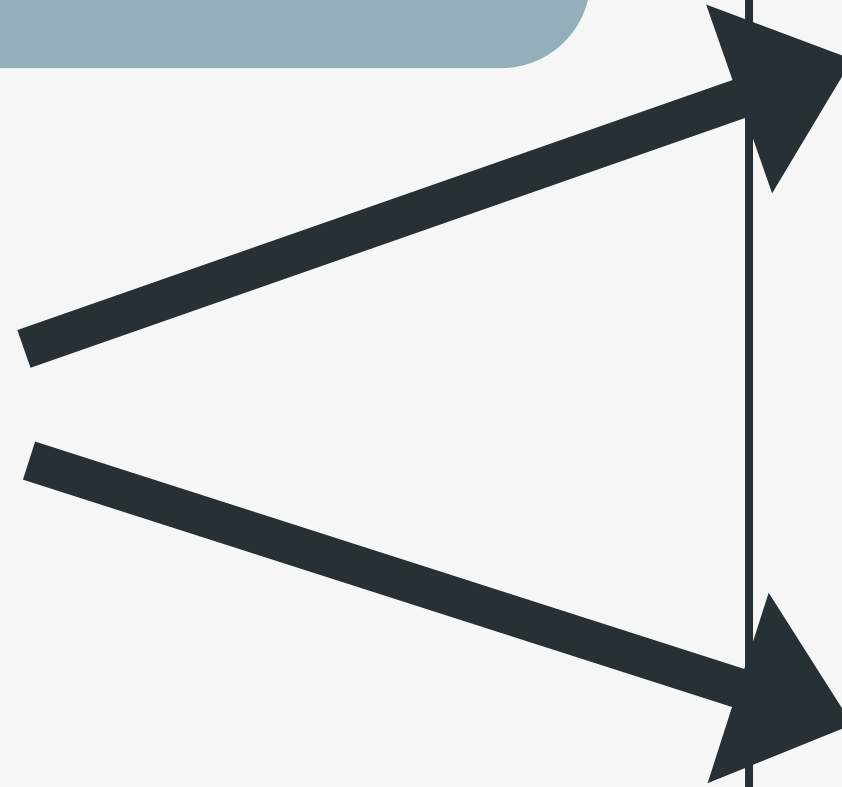


## 05 프로젝트 생성 구현



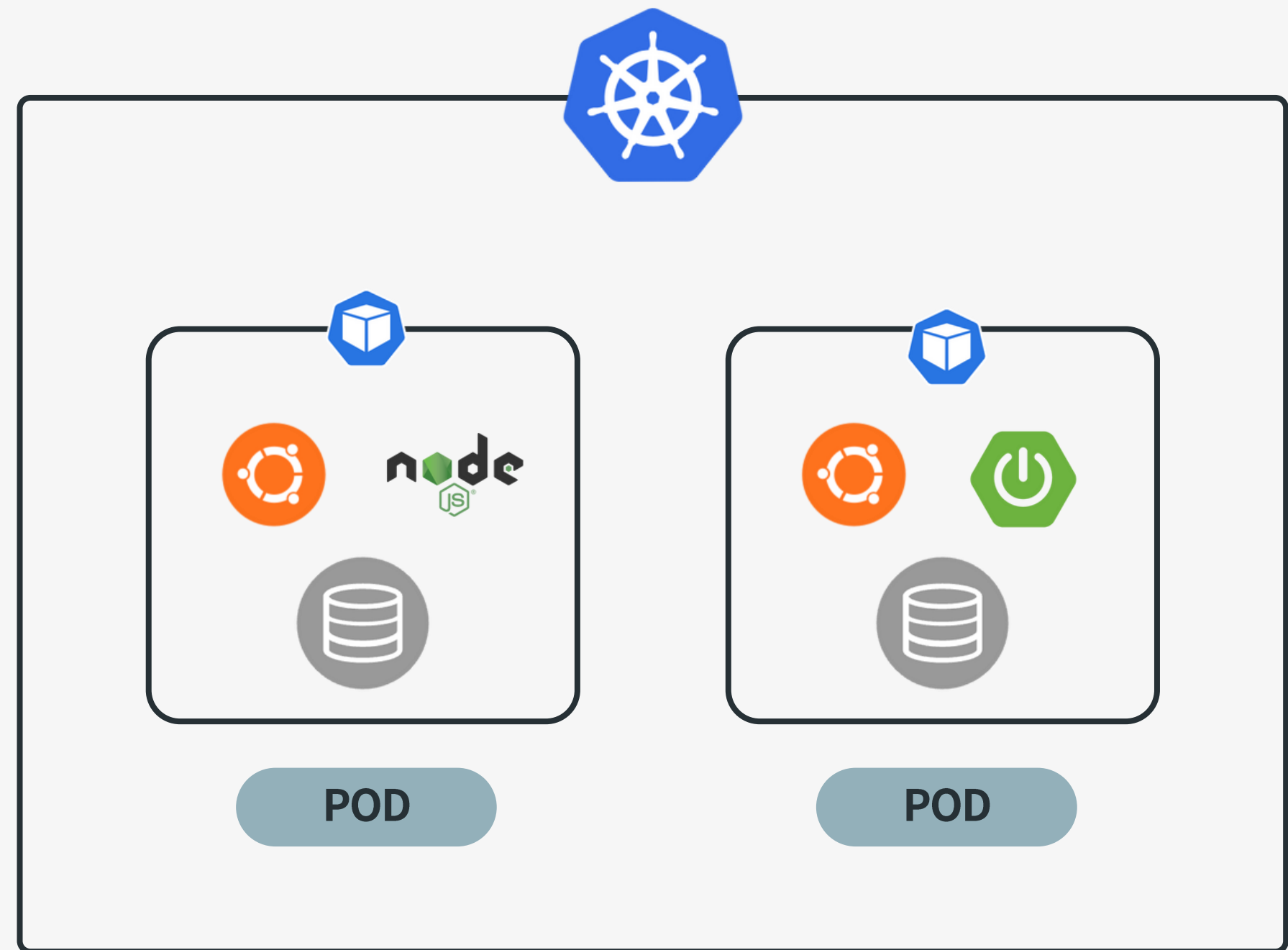


서브 도메인 라우팅



## 05 컨테이너 관리 (초기 버전)

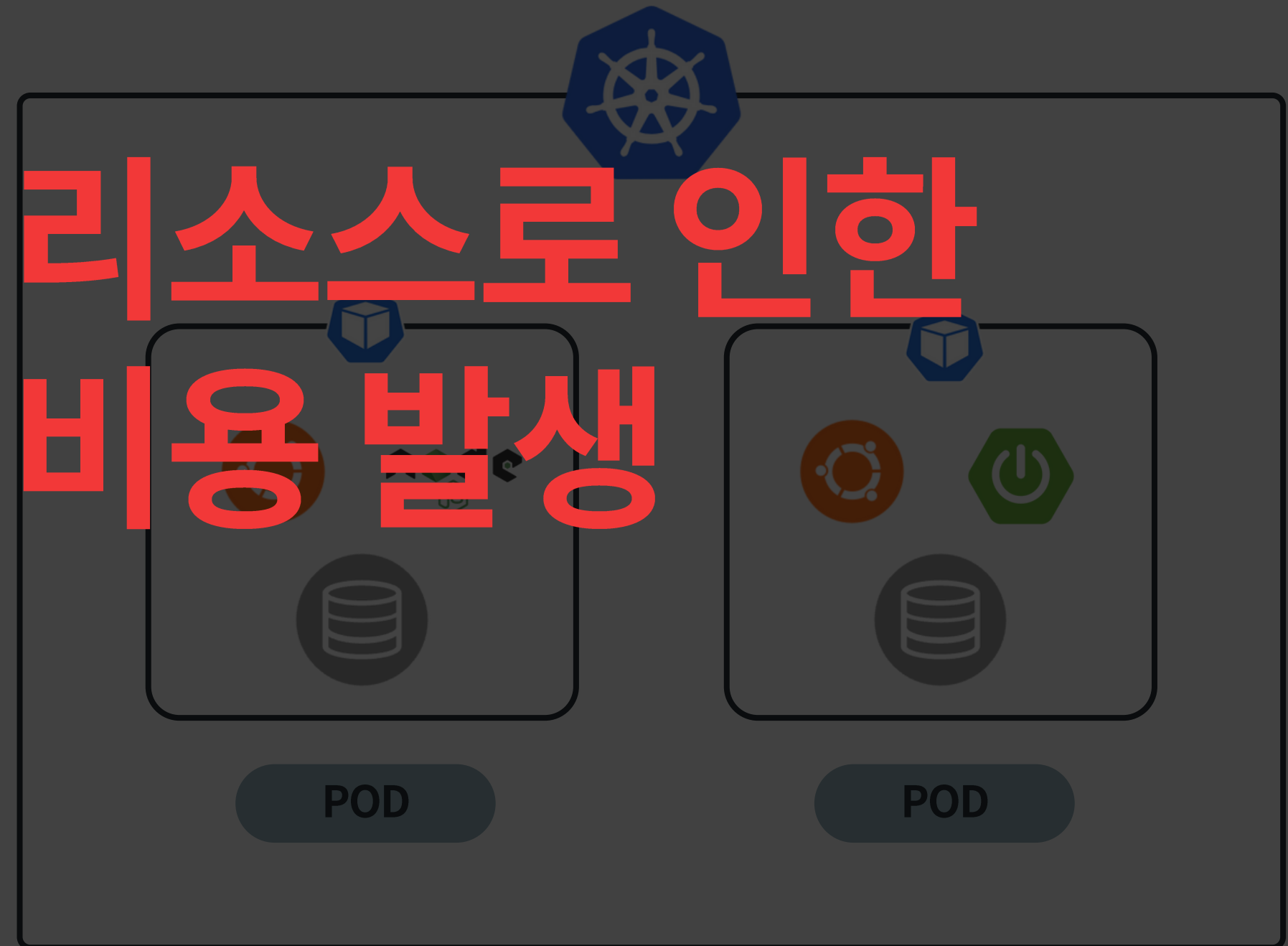
컨테이너 자원을  
관리하기 위해  
쿠버네티스와 POD 도입



## 05 컨테이너 관리 (초기 버전)

**낭비되는 리소스로 인한  
높은 비용 발생**

컨테이너 자원을  
관리하기 위해  
쿠버네티스와 POD 도입



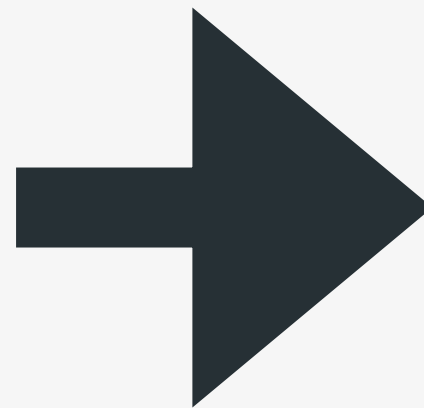
## 05 컨테이너 관리 아이디어





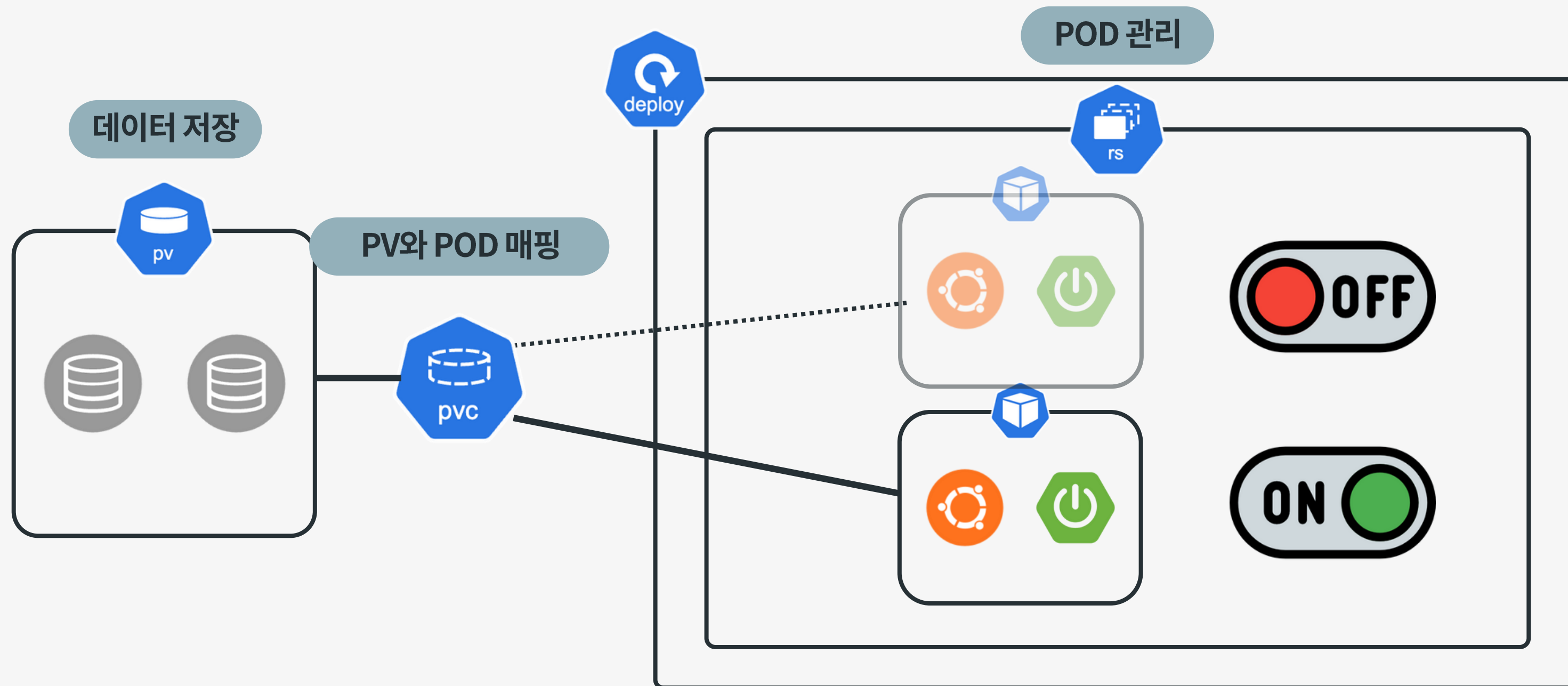
## 05 여전한 문제

~~POD는 기본적으로  
생성, 삭제, 스케일링만 제공~~



Deployment  
Replica Set (RS)  
+  
Persistent Volume (PV)  
Persistent Volume Claim (PVC)

## 05 컨테이너 관리 (개선 버전)

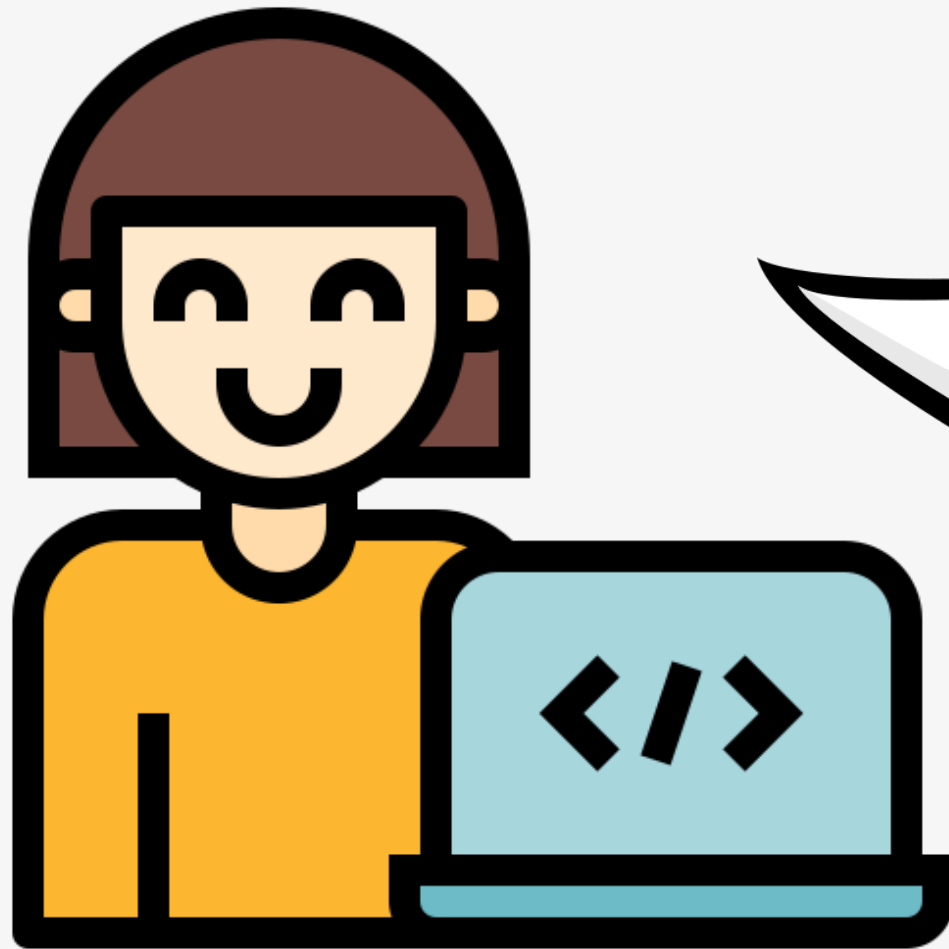


## 06 실사용 후기



팀 프로젝트 시, 하나의 코드로 같이 작업을  
하기 때문에 병합오류가 발생하는  
번거로움이 사라집니다.

## 06 실사용 후기



파일들을 받아서 확인하는 것보다  
배포 기능으로 빠르게 확인할 수 있어서  
채점 하는 데 시간이 단축될 것 같아요.

# Thank you

감사합니다.