

# AI기반 회의록 작성 및 배포 서비스

MVP 완료 보고서



커피마시는 쿼카 팀 / 2025.10.31

# 목차

## 01 MVP 주제 및 Squad구성

## 02 문제 정의 및 분석

## 03 솔루션정의

03-1 솔루션 방향성 정의

03-2 솔루션 탐색과 선정

03-3 비즈니스 가치

## 04 솔루션 전략 설계(DDD)

04-1 도메인 모델과 마이크로 서비스 정의

04-2 유저스토리 현황 및 서비스별 분석

## 05 Sprint 진행 과정과 Pivoting

## 06 상세 설계 및 주요화면

06-1 논리 아키텍처 설계

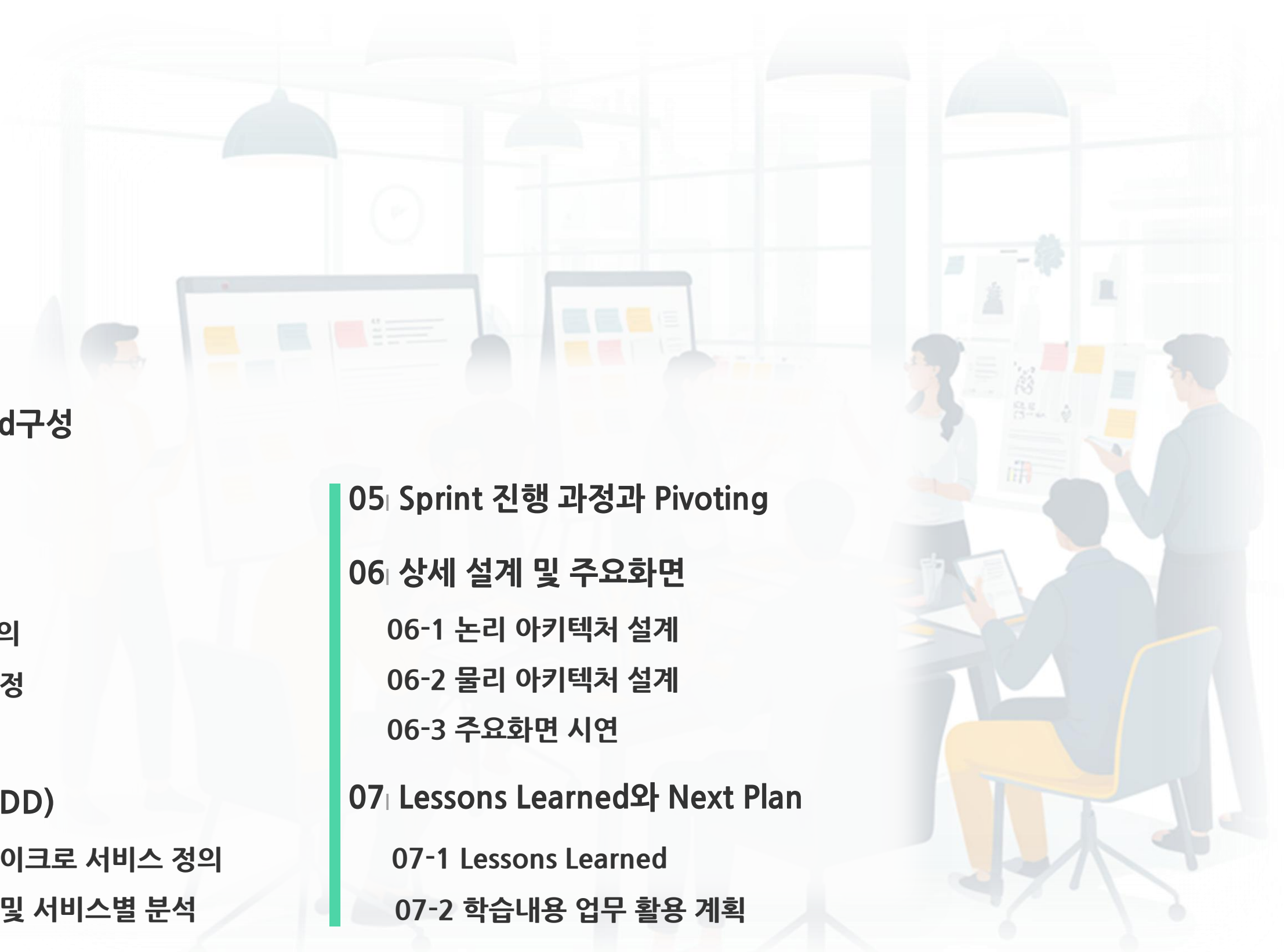
06-2 물리 아키텍처 설계

06-3 주요화면 시연

## 07 Lessons Learned와 Next Plan

07-1 Lessons Learned

07-2 학습내용 업무 활용 계획



# MVP 주제 및 배경

회의록 작성하는데 왜 야근까지 해야 하는가?

73.4%

시간 낭비 인식

직장인이 회의를 비효율적으로 평가  
(JobKorea 조사)

38%

비효율적 업무

전체 업무시간 중 비효율 비율  
(Ernst & Young Korea, 2013)

2.5시간

낭비 시간

하루 평균 불필요한 업무 소비 시간

146조원

경제적 손실

연간 손실액  
(당시 GDP의 11.6%)

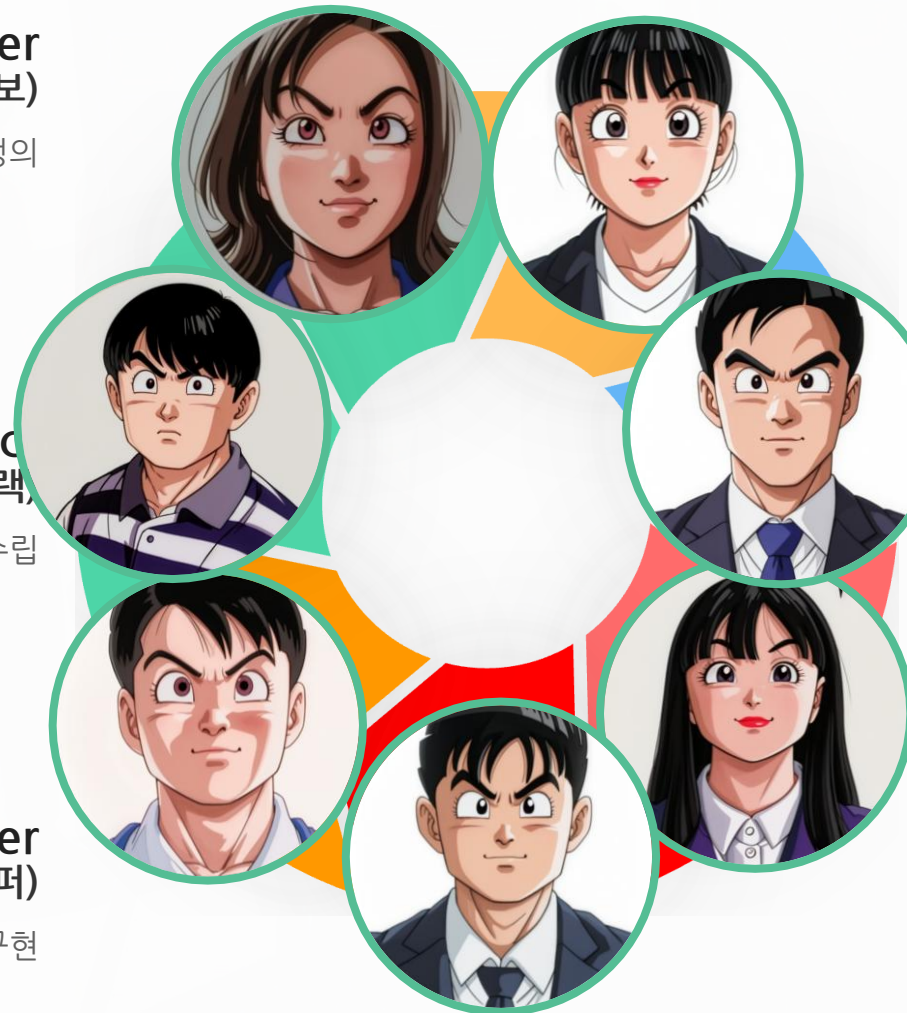
# Squad 구성

## Agentic Workflow 기반 역할 분담

**Product Owner**  
유동희 (야보)  
프로젝트 방향성 설정 및 요구사항 정의

**Architect**  
김주환 (블랙)  
시스템 아키텍처 설계 및 기술 전략 수립

**Frontend Developer**  
김종희 (페퍼)  
회의록 UI/UX 개발, 실시간 협업 인터페이스 구현



**AI Specialist & Service Planner**  
조민서 (다람지)

AI 회의록 생성, LLM 프롬프트 엔지니어링, 서비스 및 UI/UX 기획

**Backend Developer & RAG**  
전대웅 (맥심)

API 설계, DB 설계, 회의록 처리 로직 구현, RAG 처리 로직

**Scrum master & Backend Developer**  
조윤진 (쿼카)

API 설계, 비즈니스 로직 구현

**Frontend Developer & DevOps**  
문효종 (카누)

회의록 UI/UX 개발, 실시간 협업 인터페이스, CI/CD 구축

# 문제 정의 및 분석

## 현상 문제

- 1 회의록 작성에 **평균 30분~1시간 소요**된다
- 2 **전문용어** 이해 및 정확한 기록의 어렵다
- 3 회의록 **공유 지연 및 후속 조치가 누락**된다
- 4 과거 **회의 내용 검색 및 추적**이 어렵다

## 5WHY 분석

- 1 왜 회의록 작성에 시간이 오래 걸리고 정확하지 않은가?  
회의록 작성자가 업무 지식 없이 회의록을 정확하게 작성할 수 없기 때문이다.
- 2 왜 업무 지식 없이 회의록을 정확하게 작성할 수 없는가?  
회의 중 전문용어와 맥락을 이해하지 못하고 내용에 누락이 있기 때문이다
- 3 왜 회의 중 전문용어와 맥락을 이해하지 못하고 내용에 누락이 있는가?  
용어를 찾아보거나 질문할 수 없고, 이해하지 못한채로 회의록을 작성하기 때문이다.
- 4 왜 용어를 찾아보거나 질문할 수 없고, 이해하지 못한채로 회의록을 작성하는가?  
회의 중에 용어를 찾아볼 시스템이 없고 다른 사람의 업무에 방해가 될 수도 있기 때문이다.
- 5 왜 방해하는 것을 두려워하고, 누락 있는 회의록으로 남는가?  
업무방해, 저평가 등의 두려움에 의해 심리적 안전감이 부족하고, 회의 후 회의록은 제대로 검토되지 않기 때문이다.



# 솔루션 방향성 정의

## Needs Statement (핵심 요구사항 정의)

업무 지식이 없어도 정확하고 부담없이 회의록을 작성하고 공유하기 위해, 팀원은 회의 중 전문용어를 쉽게 찾고 회의록을 함께 검증하는 협업적 회의록 작성 경험이 필요.

### 업무 미숙자도 실시간 이해

회의 흐름 방해 없이 모르는 용어나 이전 회의내용 즉시 확인

### 함께 검증

회의 종료 즉시 회의록을 함께 확인하고 수정하여 정확성 향상

### 협업적 경험

혼자만의 부담이 아닌, 함께 만들어가는 협업 프로세스



# 솔루션 탐색과 선정

아이디어 우선순위 평가를 통한 솔루션 선정

## ◇ 우선순위 평가 기준

- 사용자 가치: 업무 지식이 없는 사용자도 정확한 회의록 작성 가능 여부
- 기술적 실현 가능성: MVP 기간 내 구현 가능 여부 (3개월)

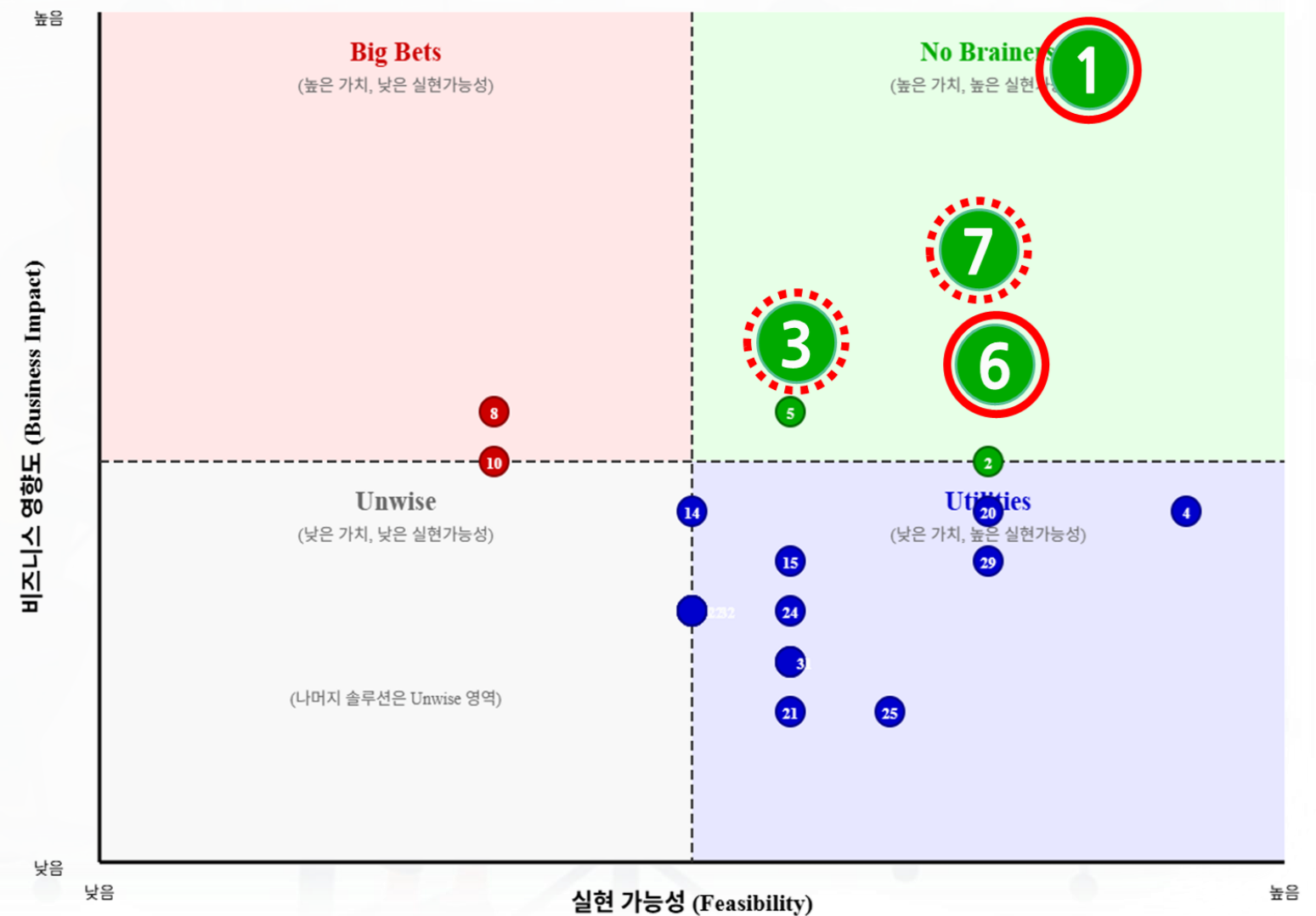
**1** AI 기반 실시간 회의 음성 인식 및 자동 회의록 작성 시스템

**6** AI 기반 회의내용 실시간 요약 및 정리

**7** 회의록 기반 Todo 및 일정 자동관리

**3** 실시간 용어/지식 자동 설명 및 RAG 지식 증강 시스템

솔루션 우선순위 평가 매트릭스



# 비즈니스 가치

## 사용자 관점

시간 단축  
↓  
작성시간 **80%** 단축

실시간 협업 및 피드백  
↓  
**실시간 검증** 및 수정

학습시간 단축 및 지식 지원  
↓  
**전문용어 즉시 해설**  
RAG 기반 과거 문서 자동 참조

## 기업 관점

회의 준비 시간 50% 감소 (이전 회의록 자동 연결 및 맥락 제공)

**연 1.4억원** 인건비 절감 (500여명 규모, 연간 4,800시간 절약)

신입 직원 온보딩 기간 40% 단축 (과거 회의록 학습 활용)

지식 자산화로 업무 생산성 **30%** 향상 (리서치결과 63%가 미활용)

직원 참여도 유도로 기업 성과 상승 및 이직률 감소

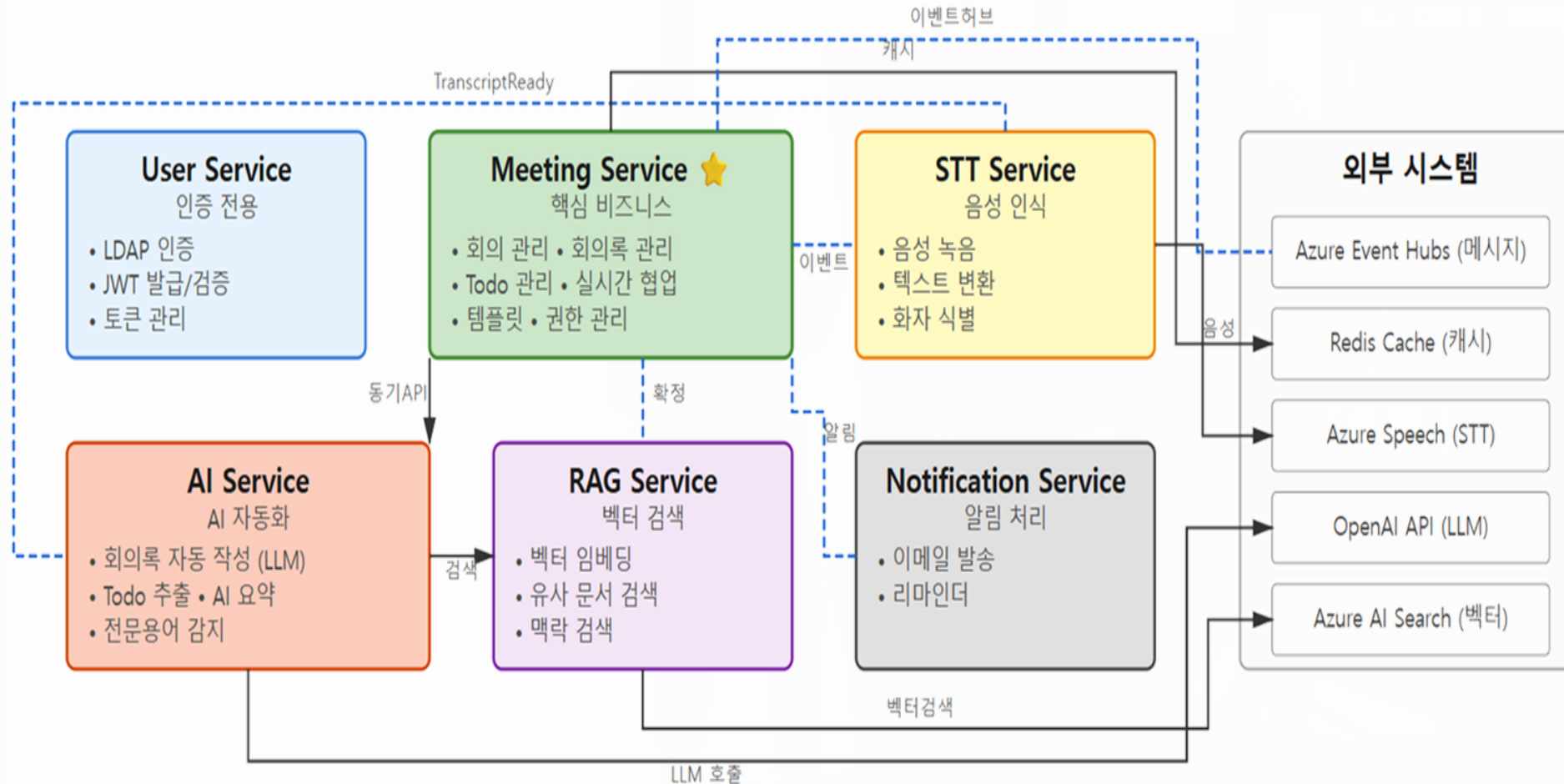
회의 결정사항의 실제 실행률 **90%** 이상 보장

(경영진-직원 간 신뢰 구축, 61%가 더 나은 문화를 위해 이직)



# 도메인 모델과 마이크로 서비스 정의

# Domain-Driven Design (DDD) 기반 설계



범례

→ 동기 호출  
--- 이벤트 (비동기)

## 핵심 설계 원칙

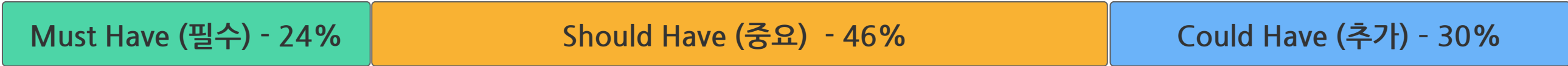
- 서비스 독립성: 각 서비스는 독립 DB 보유
- 이벤트 기반 통신: Azure Event Hubs를 통한 Pub/Sub (비동기)
- 선택적 동기 통신: Meeting → AI 즉시 응답 필요 시 REST API (Partnership)

## 마이크로서비스 구성

- 1 User**  
사용자 인증 (LDAP, JWT)
- 2 Meeting**  
회의, 회의록, Todo 통합 관리,  
안건별 검증완료 및 잠금 (Last Write Wins)
- 3 STT**  
음성 스트리밍, 실시간 음성-텍스트 변환
- 4 AI**  
회의록 자동화, Todo 추출, 안건별 AI 요약, RAG 서  
비스 연동
- 5 RAG**  
용어집 검색 (PostgreSQL+pgvector), 관련자료 검  
색 (Azure AI Search), 회의록 유사도 검색 (Vector  
DB)(Python/FastAPI 독립 서비스)
- 6 Notification**  
이메일 알림 (회의 시작, 회의록 확정, 참여자 초대)

# 유저스토리 현황 및 서비스별 분석

## MoSCoW 우선순위별 분포



## 서비스별 상세 분석

### Meeting 서비스 (42.4%)

가장 많은 기능 보유

- **Must:** 회의예약, 회의시작, 회의종료, 생성자 권한
- **Should:** 회의수정, 템플릿, 회의록확정, 입/퇴장
- **Could:** 진행중 초대, AI 메모

### AI 서비스 (18.2%)

차별화 핵심

- **Must:** 회의록 자동 생성
- **Should:** Todo 추출, 요약, 실시간 전문 용어감지
- **Could:** 실시간 AI 제안, 회의록 연결

### RAG 서비스 (9.1%)

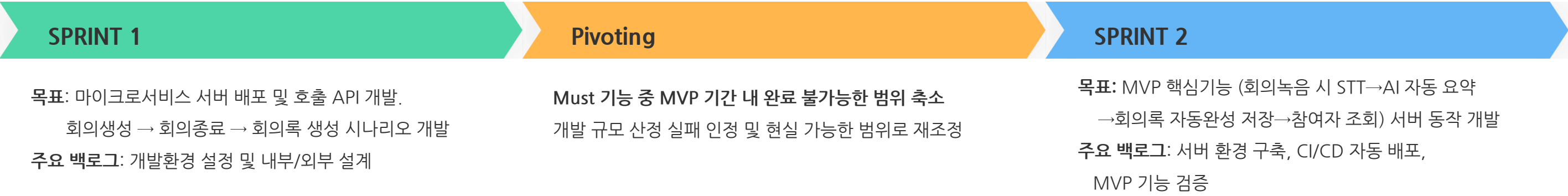
모두 Could Have

- 전문용어 검색, 맥락 기반 설명, 관련 회의록 검색
- 선택적 구현으로 차별화

핵심 MVP는 “AI 기반 회의록 자동 작성”에 집중되어 있으며, Must Have 8개 중 5개가 회의 진행과 음성 처리에 집중되어 있습니다. Should Have가 45.5%로 가장 많은데, 이는 사용자 경험 완성도를 위한 기능들(목록 조회, 수정, Todo 등)이 많기 때문입니다.

# Sprint 진행 과정과 Pivoting

## Sprint 진행 사항

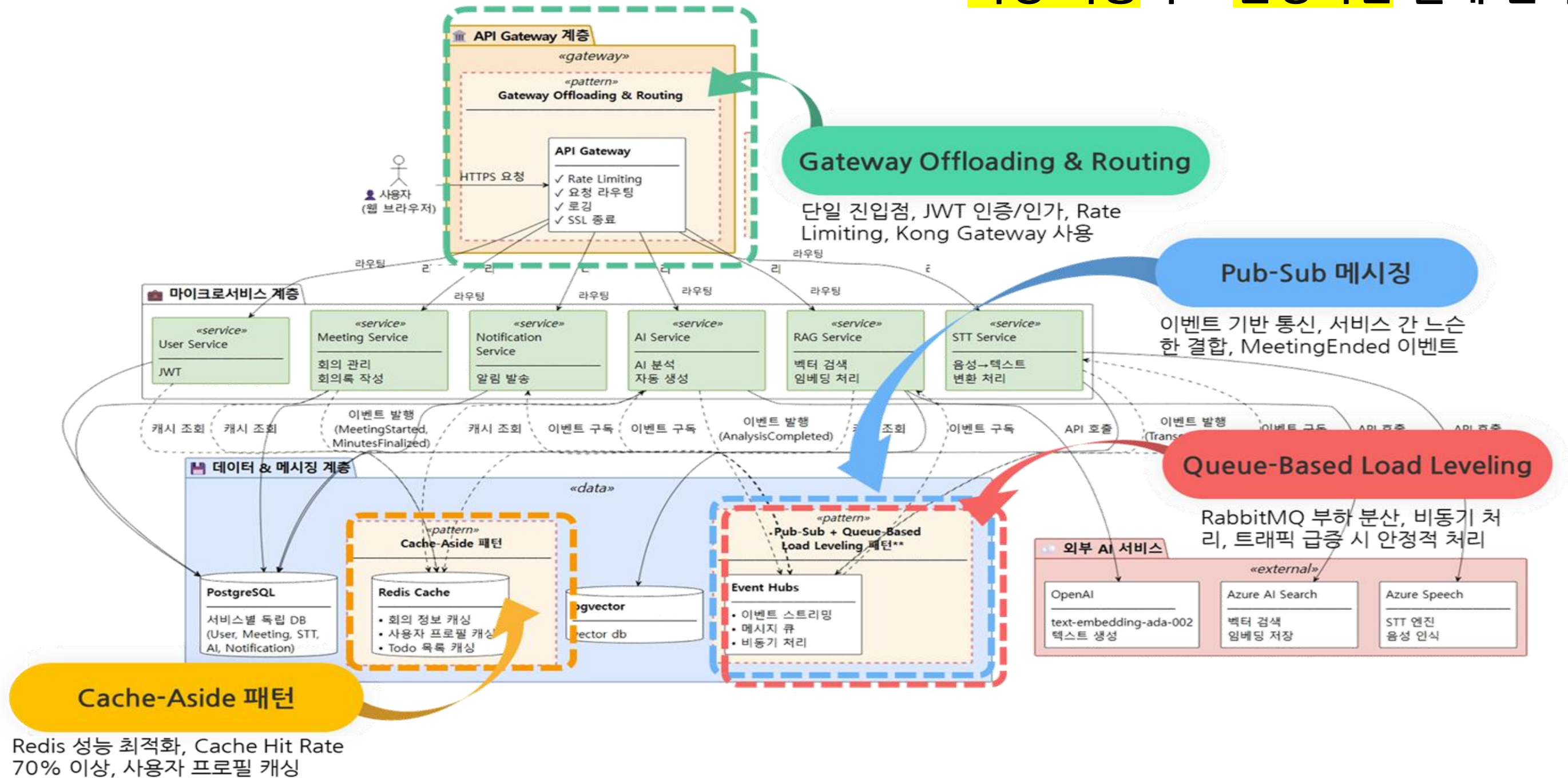


## Pivoting 포인트와 의사결정

Pivot 1: Todo 서비스 통합	Pivot 2: 실시간 동기화 방식 단순화	Pivot 3: AI 요약 재생성 속도 최적화
<p>초기 계획: Todo 별도 마이크로서비스 분리문제점: 통신 오버헤드, 데이터 정합성 복잡도 증가의사결정: Meeting Service에 Todo 관리 기능 통합결과: 개발 속도 30% 향상, API 호출 50% 감소</p>	<p>초기 계획: WebSocket 기반 실시간 양방향 통신문제점: 동시 수정 충돌 해결 복잡도, 상태 관리 어려움의사결정: Last Write Wins 정책 + 안전별 잠금 채택결과: 충돌 해결 로직 단순화, 사용자 경험 개선</p>	<p>초기 계획: GPT-4 모델 사용 (응답 시간 10-15초)문제점: 사용자 대기 시간 길어 UX 저하의사결정: 안전별 요약 GPT-3.5-turbo 사용 + 프롬프트 최적화결과: 응답 시간 2-5초 단축, 사용자 만족도 향상</p>

# 논리 아키텍처 설계

확장 가능하고 안정적인 설계 전략





# 물리 아키텍처 설계

**Backend**  
Java 17 + Spring Boot 3.2  
엔터프라이즈급 안정성 보장

**Frontend**  
React 18 + TypeScript  
실시간 협업 UI 최적화

**AI Service**  
Python 3.11 + CLAUDE  
AI/ML 생태계, 빠른 응답

**DB**  
PostgreSQL 15  
구조화 데이터, ACID 보장

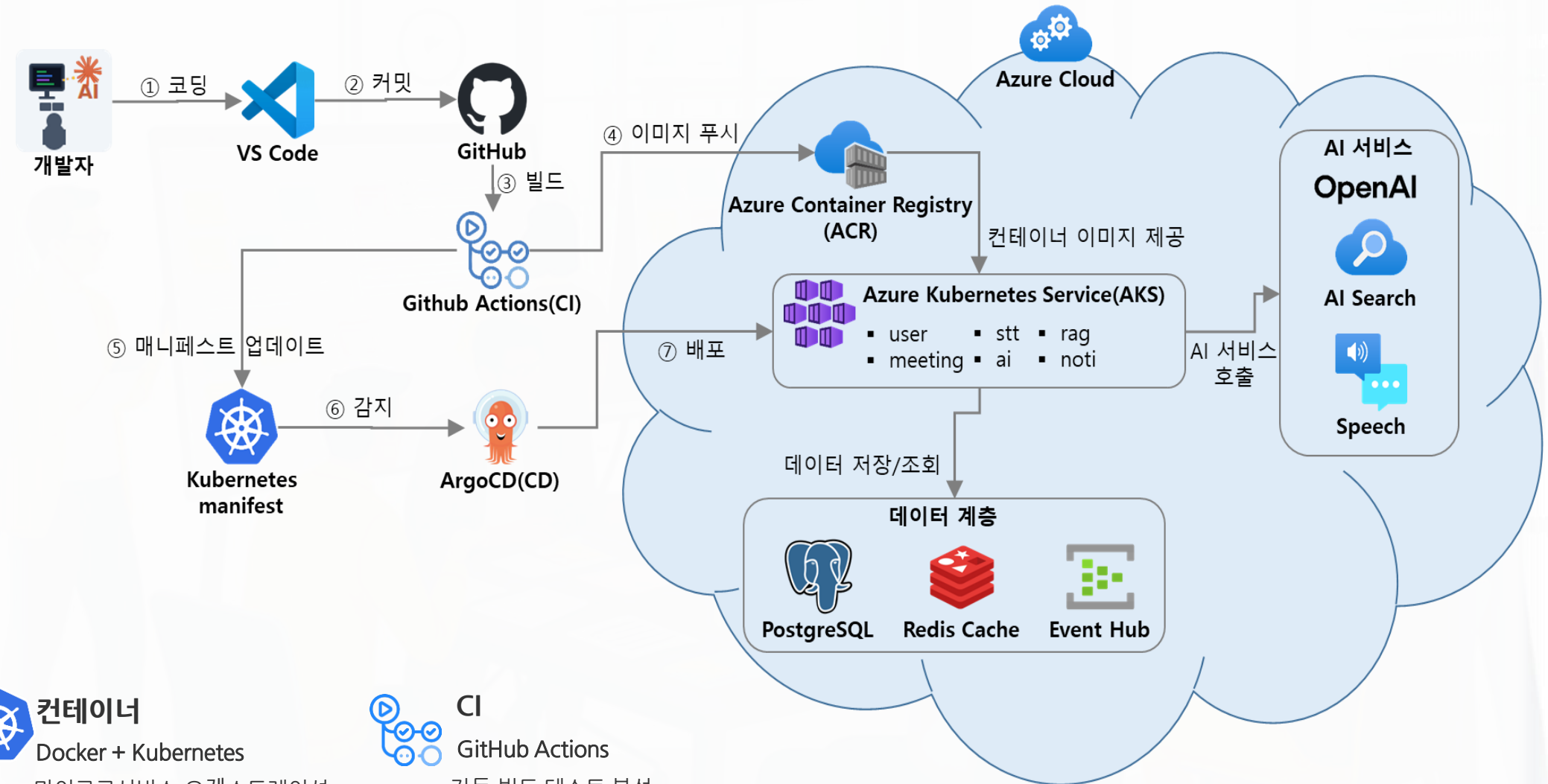
**캐시**  
Redis  
실시간 세션 관리 지원

**컨테이너**  
Docker + Kubernetes  
마이크로서비스 오케스트레이션

**MQ**  
Azuer Event Hub  
AI 비동기 처리, 이벤트 기반

**CI**  
GitHub Actions  
자동 빌드·테스트·분석

**CD**  
ArgoCD  
GitOps 기반 자동 배포



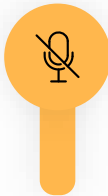


# MVP 시연 - 주요 기능



## 회의 예약 및 시작

회의 생성 및 템플릿 선택, 회의 시작 및 실시간 음성 녹음



## 회의 진행 중

실시간 STT 변환, AI 주요 내용 제안, 전문용어 자동 감지 및 설명, 관련 회의록 자동 연결



## 회의 종료 및 회의록 생성

회의 종료, AI 자동 회의록 작성 (안건별 요약), Todo 자동 추출



## 회의록 수정 및 확정

AI 요약 재생성 (2-5초), 안건별 검증 완료, 회의록 최종 확정

# 배운 점과 소감



AI를 통해 생산성은 10배이상 높일 수 있지만, 아직까지는 잘못된 방향으로 가지 않게 검증하는 사람의 역할도 중요한 것 같습니다.



5주간 좋은 강사님과 좋은 동료들과 같이 AI활용 및 적용에 대해 많이 배웠습니다



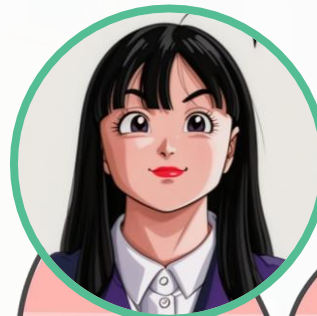
AI는 거품이 아니다!  
AI의 효용성을 확인할 수 있었고, 의사결정하고 검토할 수 있는 역량이 중요하다는 것을 배웠습니다.



Agile 개발방법에 대해서 몸소 느낄 수 있었으며, 현업에서 프로젝트 참여를 한다면 적응하는데 어렵지 않을 것 같습니다.



이번 교육을 통해 AI가 제안하는 결과를 그대로 믿기보다, 프로젝트의 목표, 방향을 명확히 이해할 때 AI를 개발 동료로 활용할 수 있다는 점을 깨달았습니다.



현업에서 AI를 어떻게 활용할 수 있는지에 대해 생각할 수 있었고, 이와 더불어 개발 역량도 잘 쌓아야 될 것 같습니다.



뽕뽕한 듯 순진(?)한 AI! 잘 다뤄서 사용하면 일상 업무에서도 잘 활용할 수 있겠다!

# 학습내용 업무 활용 계획

Agentic Workflow  
명확한 역할 분담과 자율성으로 전문성을 극대화하고  
**빠른 의사결정**



DevOps 자동화  
GitOps·CI/CD 자동화로  
**표준화**된 **배포**와 휴먼에러 최소화 가능



AI/LLM 활용  
RAG와 프롬프트 엔지니어링으로  
**정확**하고 **일관된** AI 응답 구현



유저스토리 기반 개발  
MoSCoW 기반으로 **핵심가치**에 **집중**해  
사용자 중심 제품을 개발

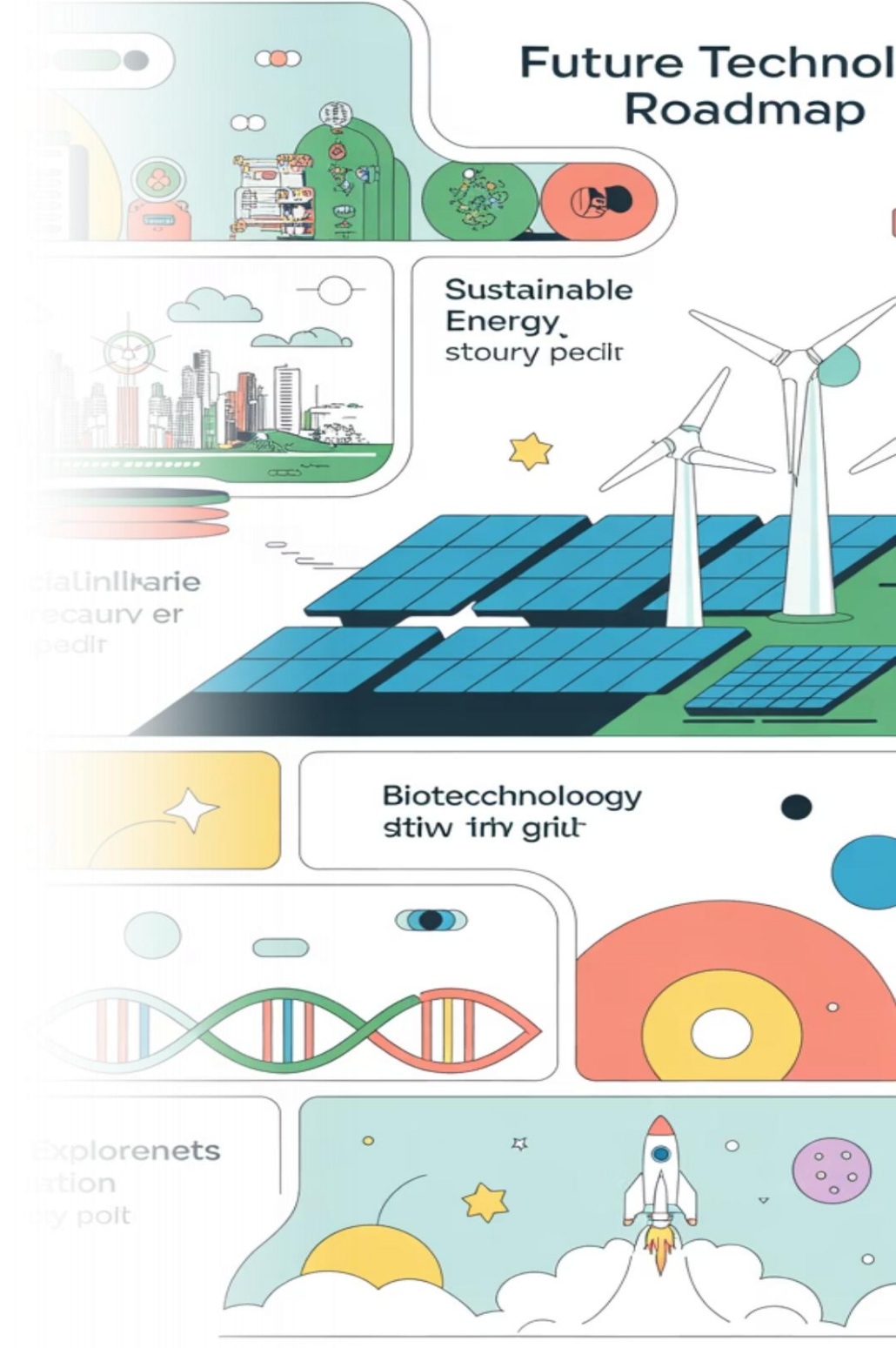


빠른 Pivoting  
**데이터 기반 회고**로 개선·전환을  
신속히 실행해 시장 적합성 강화



MSA 아키텍처  
**느슨한 결합**, 클라우드 패턴 표준화로  
확장형 유연 시스템 구축

## 07| Lessons Learned와 Next Plan







**감사합니다.**