

# Aixary 캡스톤 계획발표

AI x Diary = Aixary

모바일융합공학과

20221017 박건상 · 20227002 이선헤

2025년 9월

# 프로젝트 개요 및 목표

## 📖 Aixary (AI x Diary) - 대화형 AI 기반 일기 자동 생성 플랫폼

- 💡 사용자와 AI의 **자연스러운 대화**를 통한 일기 작성 환경 제공
- ♥️ 감정 분석과 시각화로 **사용자의 정서 변화** 추적 및 통찰력 제공

## 🎯 핵심 목표

- ✓ 일기 작성의 **진입장벽 낮추기** - 대화를 통한 자연스러운 기록
- ✓ 감정 관리의 **효율성 증대** - AI 기반 감정 패턴 분석 및 인사이트
- ✓ 사용자 경험 **개인화 극대화** - 맞춤형 피드백과 추천 서비스

## 차별화 포인트

- ★ 단순한 AI 대화가 아닌, **일기 작성과 감정 관리를 통합한** 라이프스타일 플랫폼
- ★ 지속적인 **자기 성찰과 정서적 건강**을 돕는 AI 동반자 역할 수행



# 주요 기능

## 일기 자동생성

일상 대화만으로 AI가 사용자 말투와 스타일을 반영한 일기 자동 작성

## 감정 분석 및 맞춤형 피드백

텍스트 감정 분석으로 기분 상태 파악, 개인화된 조언과 위로 제공

## 감정 달력 및 리포트

월별/주별 감정 트렌드 시각화 분석, 정기적 심리 상태 리포트 제공

## 확장된 공유 기능

AI 생성 일기 요약 이미지, 감정 기반 퀴즈 및 인사이트 공유 기능



# 기술 스택 및 역할분담

## 프론트엔드

모바일 앱 인터페이스 및 사용자 경험

 React Native 크로스 플랫폼

 UI/UX 디자인 직관적 인터페이스

## 백엔드

서버 및 API 구현

 Spring Framework API 서버

 사용자 인증 및 데이터 보안

## 데이터베이스

데이터 저장 및 관리

 관계형 DB + 인덱싱

 MVP 단계 단순 구조 적용

## AI 엔진

일기 생성 및 텍스트 처리

 OpenAI GPT-4O 대화 처리

### 박건상 (20221017)

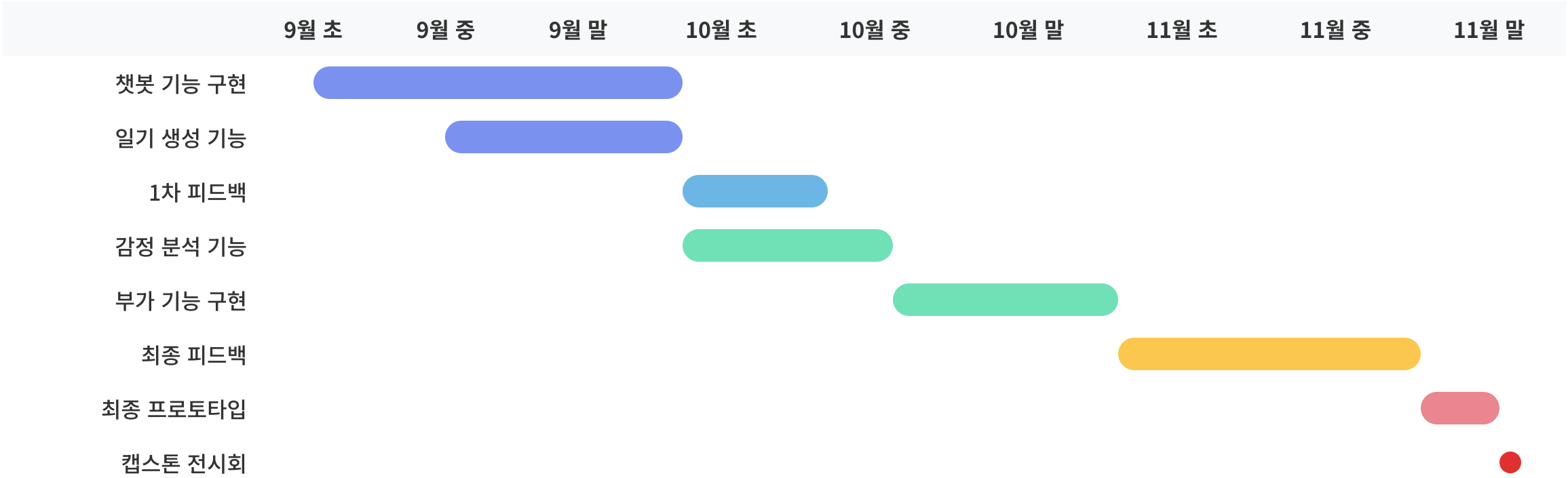
- 프론트엔드 개발 (React Native)
- 모바일 UX/UI 디자인 및 구현
- 클라이언트 상태 관리 및 API 연동

### 이선헌 (20227002)

- 백엔드 서버 개발 (Spring)
- GPT-4O API 연동 및 프롬프트 설계
- 데이터베이스 설계 및 API 구현



# 개발 일정(간트차트)



\* 주간 단위로 진행 상황 점검 및 회의 진행 중

# 연구내용 1: 창업 경험에서의 깨달음

💡 기술이 목적이 되어서는 안된다

› 사용자 경험에 기여하는 가치가 중요

## 사용자 중심 사고의 4가지 원칙

### ⚡ 체감 가치 우선

즉각적인 개선 체감 (속도, 편의성, 단순함)

### 👤 직관적인 경험

단순하고 이해하기 쉬운 사용자 흐름 설계

### 🔄 빠른 피드백 루프

신속 제작 → 사용자 반응 → 즉시 개선 순환

### ✂ 경험 중심 기술 선택

사용자 경험에 직접 기여하는 기술 우선

### 핵심 교훈

“ 사용자 중심으로 사고하는 엔지니어가 되어야 한다.

“ 신속 개발, 피드백 반영, 불필요한 복잡성 제거



# 개발 과정에서의 현실적 결정

프로젝트를 진행하며 내린 중요한 결정과 이유

## 복잡한 데이터베이스 대신 단순화

### 처음 생각

고급 검색과 추천을 위한 특별한 데이터베이스(벡터DB) 사용 계획

### 발견한 문제

현재 규모에서는 너무 복잡하고 비용과 관리가 부담됨

→ 초기 버전에는 과한 기술



### 최종 결정

간단한 데이터베이스와 검색 기능으로 시작하고, 나중에 필요하면 고급 기능 추가

→ 사용자가 많아지면 업그레이드

## 자동 기억/망각 기능 보류

### 처음 생각

AI가 중요한 내용은 기억하고 덜 중요한 내용은 자동으로 잊게 하는 기능

### 발견한 문제

어떤 내용을 기억하고 잊을지 판단하기가 어렵고, 개인정보 문제 발생 가능

→ 사용자가 불안해할 수 있음



### 최종 결정

사용자가 직접 중요한 내용을 저장하거나 삭제할 수 있게 단순화

→ 사용자에게 더 많은 제어권 제공

## 웹에서 모바일 앱으로 전환

### 처음 생각

웹 기반(Next.js)으로 개발하여 관리와 배포를 쉽게 하려 함

### 발견한 문제

일기는 '언제 어디서나 손에 있는 모바일'이 핵심

→ 인터넷 없이도 사용 가능해야 함



### 최종 결정

모바일 앱으로 전환(React Native 사용)

→ 더 편리한 모바일 경험 제공

# 결론

## 아키텍처 변화 비교

### 변경 전 (1학기 계획)

Next.js → 벡터DB → 복잡한 기억 모듈 → LLM 호출

VS

### 변경 후 (MVP)

React Native → 간단한 DB + 메타 인덱스  
LLM 서버 사이드 호출 → 사용자 편집/핀 기능

## 핵심 교훈

### 💡 실용성 중심 설계

이론적으로 완벽한 아키텍처보다 **실제 구현 가능한 범위**에 집중한 설계가 효과적

### 🚀 단계적 확장 전략

초기에는 **간단한 구조**로 시작하고, 사용자 피드백과 데이터에 기반하여 **단계적 고도화**

### 👤 사용자 제어 우선

**자동화된 복잡한 기능**보다 **사용자가 직접 통제할 수 있는 기능**이 초기 채택과 만족도 향상에 효과적

### “ 결론:

아키텍처는 **기술적 완성도**보다 **사용자 경험과 실제 가치 전달**에 초점을 맞춰 발전시키는 것이 중요



# 향후 계획

## ✔ MVP 목표 달성 계획

캡스톤 전시회까지 대화형 일기 생성, 감정 분석, 캘린더, 공유 기능 등 핵심 기능이 포함된 프로토타입 완성

## ☰ 기능 우선순위 설정

1순위: 대화형 일기 생성 안정화, 2순위: 감정 분석 기본 기능, 3순위: 캘린더 및 공유 기능 구현

## 👥 사용자 테스트 계획

10월 중 1차, 11월 초 최종 피드백 수집 및 반영을 통한 사용자 경험 개선 집중

## 🚩 캡스톤 시연 목표

실제 사용 가능한 수준의 MVP 제품 시연으로 프로젝트 가치 입증 및 사용자 경험 검증

감사합니다!

