

----- 최종 정리 -----

<역할 분담>

1. PM - 수빈
2. ui/ux - 수빈
3. 프론트엔드 - 현정, 주환, 수빈
4. 백엔드 - 우림(+ a)
5. 병합 책임자 -
 - 최종적으로 main에 올리기 전에 기능별로 별도 레파지토리에 병합 테스트 필요
 - ex) 로그인 기능 구현 후 테스트, 게시판 구현 후 테스트, ... main에 최종적으로 병합
6. 보고서 작성 - 수빈
7. 발표 및 시연 -

<기능 구현>

* 사용자 페이지

[회원가입/로그인]

- 회원가입
- 로그인

[메인 홈페이지]

- 사용자 검색 -> 친구 팔로우 기능
- 메인 홈 프로필 사진, 소개, 타이틀 내용 변경
- 방문자수 나타내기
- (프로필에 기분 표현 / ex.기쁨, 슬픔, 졸림 등)

[카테고리]

- 다이어리 게시판

- 사진첩 게시판
- 방명록 게시판
- (일정 게시판)

[설정]

- 설정 페이지 만든 뒤 아래 기능 구현
- (사용자 신고 기능)
- (배경색 변경 기능)
- 회원 탈퇴
- 친구 목록

* 관리자 페이지

- 전체 회원 수
- (신고 접수 내역)

<개발 단계>

프로젝트의 효율적인 진행을 위해 각 기능을 프론트엔드와 백엔드로 나누어 단계별로 개발하는 것이 중요합니다. 여기서는 백엔드를 먼저 설계한 후, 프론트엔드와 백엔드를 병렬로 개발하는 접근을 추천합니다. 각 기능의 개발 순서를 구체적으로 제시합니다.

1. 기획 및 준비 단계

- 요구사항 분석: 모든 기능의 요구사항을 명확히 정리합니다.
- 데이터베이스 설계: 모든 기능에 필요한 데이터베이스 구조를 설계합니다.
(사용자, 팔로우, 게시판 등)

2. 초기 설정 및 기본 구조 개발

- 서버 설정 및 환경 구축: 서버 환경을 설정하고, 기본적인 서버 구조를 구축합니다.
- 초기 API 설계 및 구현: 회원가입, 로그인, 사용자 인증 API를 설계하고 구현합니다.

- 프론트엔드 기본 구조 작성: HTML/CSS로 기본 레이아웃을 작성합니다.

3. 사용자 인증 및 프로필 관리

- back
 - 회원가입, 로그인 API 완성 및 보안 설정.
 - 회원탈퇴 API 구현.
 - 사용자 프로필 및 설정 관련 API 구현 (프로필, 타이틀 변경 등).

- front
 - 회원가입 및 로그인 페이지 작성.
 - 회원탈퇴 페이지 작성.
 - 프로필, 타이틀 변경 기능 구현.

4. 친구 및 팔로우 기능

- back
 - 팔로우 기능 관련 API 구현.
 - 친구 목록 API 구현.

- front
 - 팔로우 버튼 및 인터페이스 구현.
 - 친구 목록 페이지 작성.

5. 게시판 기능 개발

- back
 - 다이어리, 사진첩, 방명록 게시판 API 설계 및 구현.
 - 일정 게시판 (선택사항) 및 투데이 API 구현.

- front
 - 각 게시판의 페이지 및 UI 구현.
 - 게시물 작성, 수정, 삭제 기능 추가.

6. 추가 사용자 기능

- back
 - 사용자 신고 API 구현.
 - 배경색 변경 API 구현.
- front
 - 사용자 신고 인터페이스 구현.
 - 배경색 변경 기능 구현.

7. 관리자 페이지 개발

- back
 - 전체 회원 수 조회 API 구현.
 - 신고 접수 내역 API 구현.
- front
 - 관리자 대시보드 페이지 작성.
 - 회원 수 및 신고 접수 내역 표시 기능 구현.

8. 테스트 및 통합

- 테스트: 각 기능별로 유닛 테스트, 통합 테스트를 진행하여 버그를 수정합니다.
- 통합: 프론트엔드와 백엔드 통합 테스트를 진행하여 전체 시스템이 원활히 동작하는지 확인합니다.

9. 배포 및 유지보수

- 배포: 최종 코드를 서버에 배포하고, 도메인 설정 및 SSL 인증서를 적용합니다.
- 유지보수: 사용자 피드백을 수집하고 지속적으로 버그를 수정하며 기능을 개선합니다.

이렇게 단계별로 진행하면 체계적이고 효율적으로 프로젝트를 개발할 수 있습니다. 초기 단계에서 백엔드의 기본 구조를 설계한 후, 프론트엔드와 병렬로 개발하는 것이 중요합니다. 또한, 주기적인 통합 테스트를 통해 각 기능이 잘 연동되는지 확인하는 것도 중요합니다.