

[팀 프로젝트] 반려식물 소셜 플랫폼 기획안

- 함께 키우는 즐거움, '푸룻푸룻(PurutPurut)' -

팀명: 초록엄지 (Team Green Thumb)

팀장: 김코딩 외 3명

2026년 1월 28일

차 례

1	프로젝트 개요	2
1.1	기획 배경	2
1.2	프로젝트 목표	2
2	팀 구성 및 역할 분담 (R&R)	2
3	시장 분석 (SWOT)	2
4	상세 기능 명세	2
4.1	핵심 기능	3
4.2	도전 과제 (Advanced)	3
5	시스템 아키텍처 및 협업 툴	3
5.1	기술 스택 (Tech Stack)	3
5.2	협업 툴 (Collaboration)	3
6	데이터베이스 설계 (ERD Draft)	3
7	개발 일정 (6주 계획)	3
8	예상되는 리스크 및 해결 방안	5
9	마치며	5

1 프로젝트 개요

1.1 기획 배경

최근 1인 가구 증가와 함께 반려식물을 키우는 인구가 급증하고 있습니다. 하지만 혼자 키우다 보면 정보가 부족해 식물을 죽이거나, 금방 흥미를 잃는 경우가 많습니다. 저희 팀은 **"식물을 매개로 소통하고, 서로의 노하우를 공유하자"**는 취지로 4명의 팀원이 모여 이 프로젝트를 기획하게 되었습니다.

1.2 프로젝트 목표

- 생존율 향상: 물주기 알림 및 식물별 맞춤 가이드 제공
- 커뮤니티 활성화: '자랑하기' 게시판을 통해 유저 간 소통 유도
- 데이터 축적: 유저들이 올린 식물 성장 데이터를 시각화
- 팀 목표: 애자일(Agile) 방법론을 적용하여 협업 경험 쌓기

2 팀 구성 및 역할 분담 (R&R)

팀원 각자의 강점을 살려 역할을 분배했습니다. 프론트엔드 2명, 백엔드 2명으로 구성되었습니다.

이름	포지션	담당 업무
김코딩 (팀장)	Backend	DB 설계, API 서버 구축, 배포(AWS), 회의 주관
이프론	Frontend	메인 페이지, 식물 등록/관리 UI, 상태 관리(Recoil)
박디자	Frontend/Design	Figma 디자인, 커뮤니티 페이지 구현, 반응형 퍼블리싱
최서버	Backend	로그인(OAuth), 마이페이지 API, 이미지 업로드 처리 (S3)

표 1: 팀원별 역할 분담표

3 시장 분석 (SWOT)

기존 앱들이 '관리'에 집중했다면, 저희는 '소통'을 차별점으로 두었습니다.

Strength (강점) <ul style="list-style-type: none">- 커뮤니티 기능을 강화하여 유저 이탈 방지.- 식물별 맞춤형 배지 시스템 (게이미피케이션).	Weakness (약점) <ul style="list-style-type: none">- 팀원 모두가 첫 프로젝트라 경험 부족.- 초기 데이터(식물 도감) 확보가 어려움.
Opportunity (기회) <ul style="list-style-type: none">- 플랜테리어(Planterior) 시장의 지속적 성장.- MZ세대의 '인증샷' 문화와 부합함.	Threat (위협) <ul style="list-style-type: none">- 대형 플랫폼의 유사 서비스 런칭 가능성.- 서버 트래픽 급증 시 대응 능력 부족.

4 상세 기능 명세

MVP(Minimum Viable Product) 개발을 목표로 하되, 추가 기능은 우선순위를 정해 개발합니다.

4.1 핵심 기능

1. 회원가입/로그인

- JWT 기반 인증 시스템 구현
- 카카오, 구글 소셜 로그인 연동

2. 마이 가든 (식물 관리)

- 내 식물 등록 (애칭, 종류, 입양일, 물주기 주기 설정)
- D-Day 계산 및 Push 알림 기능

3. 커뮤니티 (식물 자랑)

- 사진 업로드 및 게시글 작성 (CRUD)
- 좋아요 및 댓글 기능, 태그 기능 (#몬스테라 #거실화분)

4.2 도전 과제 (Advanced)

- 실시간 인기 게시글 랭킹 시스템
- 식물 상태 진단 AI (Teachable Machine 활용 예정)

5 시스템 아키텍처 및 협업 툴

5.1 기술 스택 (Tech Stack)

분야	사용 기술
Front-end	React, TypeScript, Recoil, Styled-components, Axios
Back-end	Node.js, Express, Sequelize (ORM)
Database	MySQL (RDS), Redis (캐싱 도전)
Infra	AWS EC2, S3 (이미지 저장), Nginx

5.2 협업 툴 (Collaboration)

- **Notion**: 회의록 작성 및 기획 문서 공유
- **Git/GitHub**: 코드 형상 관리 (Git Flow 전략 사용: main, develop, feature 브랜치)
- **Discord**: 매일 오전 10시 데일리 스크럼 진행 (음성 채팅)
- **Figma**: 화면 UI/UX 디자인 공유

6 데이터베이스 설계 (ERD Draft)

관계형 데이터베이스 설계를 위해 팀원들과 밤새 토론하며 구조를 잡았습니다.

7 개발 일정 (6주 계획)

팀 프로젝트라 일정이 빡빡하지만, 스프린트 방식을 도입해 효율적으로 진행하겠습니다.

[간이 ERD 구조도]

Users (1) — Plants (N)
(id, email, nickname) (id, user_id, name, water_cycle)

Users (1) — Posts (N)
(id, ...) (id, user_id, title, content, image_url)

Posts (1) — Comments (N)
(id, ...) (id, post_id, user_id, content)

그림 1: 핵심 테이블 간의 관계도

기간	주요 활동
1주차	기획 구체화, 와이어프레임 작성(Figma), DB 설계(ERD), 깃허브 세팅
2주차	[BE] 기본 API 서버 구축 및 DB 연동 / [FE] 공통 컴포넌트 개발 및 퍼블리싱
3주차	[BE] 로그인(JWT) 및 CRUD 구현 / [FE] 페이지 라우팅 및 API 연동 시작
4주차	핵심 기능(물주기 알림, 이미지 업로드) 구현 및 연동 테스트
5주차	추가 기능(댓글, 좋아요) 개발, UI 폴리싱(디자인 다듬기), 버그 수정
6주차	최종 배포(AWS), 시연 영상 제작, 포트폴리오 문서 정리

8 예상되는 리스크 및 해결 방안

1. Git 충돌(Conflict) 문제:

- 해결: 기능별로 브랜치를 명확히 나누고, Merge 전에 팀장 코드 리뷰를 필수로 진행합니다.

2. 의사소통 오류:

- 해결: 매일 아침 스크럼을 통해 "어제 한 일, 오늘 할 일, 이슈 사항"을 공유합니다.

3. 일정 지연:

- 해결: 핵심 기능(MVP)을 최우선으로 구현하고, 부가 기능은 시간이 남을 때만 진행합니다.

9 마치며

혼자 공부할 때는 막막했지만, 팀원들과 함께라서 든든합니다. 이번 프로젝트를 통해 기술적인 성장뿐만 아니라, **함께 일하고 싶은 개발자**가 되는 것이 저희 팀의 최종 목표입니다. 꼭 완성해서 앱 스토어에는 못 올려도 포트폴리오에는 당당히 싣겠습니다!