

프로젝트 최종 제안서 - 8조

TastyHub



1. 팀 멤버 및 역할

2. 문제 정의

3. 프로젝트 목적

프로젝트 목표

4. 기술적 배경 및 선행 연구

기술적 배경

선행 연구

5. 프로젝트 범위 및 한계

범위

1. 사용자 인증 및 회원 가입

2. GPS 기반 재료 공동구매 게시글 작성 및 채팅 소통 서비스

3. 레시피 서비스

4. 사용자 상호작용 기능

한계

1. 오프라인 사용 및 접근성

2. UI/UX 측면의 사용자 경험 한계

3. 기능 제한

6. 기술적 요구 사항

Frontend

Backend

Infra

7. 예산 및 자원

8. 예상 결과

프로젝트

1. 음식에 대한 장벽의 간소화

2. 사람간의 소통

3. 경제적 효과

9. 일정 계획 - 마일스톤

240327 ~ 240330

240331 ~ 240403

240404 ~ 240406

240407 ~ 240410

240411 ~ 240413

240414 ~ 240420

240421 ~ 240427

240428 ~ 240504

240505 ~ 240511

240512 ~ 240518

240519 ~ 240525

240526 ~ 240601

240602 ~ 240608

1. 팀 멤버 및 역할

이름	김민수	임세현	박승은
학과	소프트웨어학과	소프트웨어학과	소프트웨어학과
학번	32197083	32183786	32207352
역할	- 팀 리더 Backend - Junit Test(단위) - API 문서화 - API 설계 및 개발 - query 최적화	Backend - 서버 환경 구축 - 클라우드 서버 관리 - CI/CD 구축 - API 설계 및 개발	Frontend, - UI/UX - 성능 최적화 - 크로스 브라우징 - 반응형 디자인

사용기술	Java, Spring, JPA, Query DSL, MySQL, Security, Rest Docs, docker, git action, jenkins, nginx	Java, Spring, JPA, MySQL, EC2, S3, RDS, docker, jenkins	React, Typescript, Css, Redux, Sass, vercel, Prettier
------	--	---	---

2. 문제 정의

사회가 발전하면서 다양한 분야의 발전이 이루어졌다. 그에 따라 음식에 관한 분야도 많은 발전을 하였고, 이전과 달리 많은 음식에 대한 정보를 얻을 수 있는 환경이 마련되었다. 이때 음식에 관한 식당에 대한 정보가 많아지고 그에 따라 사람들의 움직임도 달라졌다. 유명한 음식점에서 식사를 하는 사람들도 존재하지만, 이에 맞춰서 여러 정보들을 바탕으로 다양한 레시피가 등장 및 공유로 음식을 직접 해먹는 사람들도 존재한다. 다만 레시피마다 과정 및 시간이 편차가 심하기에 조리를 포기하는 경우도 존재한다. 방대한 레시피와 다양한 재료들로 인한 문제점 및 고민사항이라 생각하고 정의를 내리게 되었다.

3. 프로젝트 목적

프로젝트 목표

- 지역 주민 간 식재료 공동 구매 활성화
 - 공동 구매 플랫폼 개설 - 온라인 플랫폼을 통해 지역 주민들이 함께 식재료를 구매할 수 있는 시스템을 구축한다. 사용자들은 원하는 제품을 함께 구매하여 저렴한 가격에 물건을 살 수 있다.
- 레시피 공유 및 소통
 - 온라인 레시피 공유 플랫폼 운영 - 사용자들이 자신의 레시피를 업로드하고 공유할 수 있는 온라인 플랫폼을 구축한다. 레시피 검색, 평가 및 댓글 기능을 제공하여 소통을 촉진한다.
- 지역 경제 활성화
 - 각 지역의 판매 업체, 생산자들과 협력하여 지속 가능한 농업 및 식재료 이용을 촉진한다. 유기농 또는 지역 특산물을 중심으로 한 프로모션을 진행할 수 있다.

4. 기술적 배경 및 선행 연구

기술적 배경

- 사용자 인증 및 권한 관리
 - 사용자들이 웹 사이트에 가입하고 로그인할 수 있어야 한다. 이를 통해 사용자의 신원을 확인하고 웹 사이트의 다양한 기능을 활용할 수 있다. 또한 사용자의 역할 및 권한을 관리하여 사용자가 해당 커뮤니티에서 적절한 활동을 할 수 있어야 한다.
- 프로필 관리
 - 사용자들이 자신의 프로필을 만들고 관리할 수 있어야 한다. 프로필은 사용자의 정보, 활동 기록, 프로필 사진 수정, 동네 위치 변경과 더불어 프로필 삭제까지 가능해야 한다.
- 게시물 및 댓글 관리
 - 사용자들이 레시피를 공유하고 댓글을 달 수 있어야 한다. 이를 위해서는 게시물과 댓글을 저장하고 관리하는 기능이 필요하다. 사용자들이 자신의 레시피를 게시하고, 다른 사용자들이 해당 레시피에 대한 댓글을 작성할 수 있도록 해야 한다. 게시물과 댓글은 시간순으로 정렬되어 편안하게 찾을 수 있어야 한다.
- 검색 및 필터링
 - 사용자들이 다른 사용자가 공유한 레시피를 검색하고 필터링할 수 있어야 한다. 사용자가 검색창에 키워드를 입력하면, 해당 키워드와 관련된 레시피를 효율적으로 찾아줄 수 있는 검색 엔진이 필요하다. 또한, 사용자들이 검색 결과를 시간순과 같은 기준으로 필터링할 수 있도록 하는 기능도 중요하다. 이를 통해 사용자들이 원하는 조건에 맞는 레시피를 쉽게 찾을 수 있다.
- 사용자 경험 개선
 - 사용자들이 웹사이트를 쉽게 이용할 수 있도록 UI/UX 디자인에 신경을 써야 한다. 직관적이고 사용하기 쉬운 인터페이스를 제공하여 사용자들이 원하는 정보를 빠르게 찾을 수 있도록 해야 한다.
 - 당근 마켓의 UI의 경우 거래에 필요한 핵심 요소들로 최소화 하여 복잡성을 줄이고 단순 명료한 아이콘 디자인으로 시각적인 측면에서 직관성을 높여 사용자들에게 보기 쉽고 이용하기 편한 형태로 서비스를 제공한다.

선행 연구

- 본 프로젝트와 관련된 기존 연구 및 프로젝트는 다음과 같습니다.
 - **식재료 및 음식 플랫폼:** '배달의민족', '요기요' 등 주로 음식 주문과 배달에 특화된 플랫폼으로, 사용자들이 음식을 주문하고 배달 받는 데 중점을 둔다.
 - **레시피 공유 어플 / 플랫폼:** 네이버 요리(레시피), 여러가지 요리 레시피 어플리케이션은 사용자들의 레시피를 공유하고 다른 사용자들과 음식에 대한 정보를 교환하는 데 중점을 두고 있다.
 - **지역 간 소통 플랫폼:** '네이버 카페', '번개장터', '당근마켓' 등 지역 내 이웃들간의 소통과 거래를 위해 사용되는 플랫폼으로, 사용자들이 지역 정보 및 물품 거래 등 다양한 활동을 할 수 있다.

본 프로젝트는 위 기존 플랫폼의 장점을 취하면서, 지역 특화 및 소통 기능을 강화하여 요리와 관련된 서비스를 제공할 목적으로 진행된다. 이를 통해 사용자들이 지역 내 다양한 요리 문화를 공유했고 소통할 수 있는 플랫폼을 구축하고자 한다.

5. 프로젝트 범위 및 한계

범위

1. 사용자 인증 및 회원 가입

- 사용자 인증 및 회원 가입을 통해 유저 서비스 제공
- 범위 외: OAuth 2.0을 통한 소셜 로그인 기능은 포함하지 않음

2. GPS 기반 재료 공동구매 게시글 작성 및 채팅 소통 서비스

- 사용자들이 공동으로 재료를 구매하기 위한 게시글을 작성하고 참여할 수 있는 기능 제공
- 사용자의 위치 주변에서 올라오는 공동구매 글을 모아 조회할 수 있는 기능 제공
- 사용자는 재료 공동구매를 위한 인원 모집 게시글을 작성할 수 있고, 해당 공동구매에 참여하는 인원들에게 채팅 서비스 제공

3. 레시피 서비스

- 초기 출시 단계에서 1000개 이상의 레시피 제공
- 사용자가 필요 시 자신만의 레시피를 추가할 수 있는 기능 제공
- 레시피 이름이나 레시피에 포함된 재료를 통한 검색 서비스 제공

4. 사용자 상호작용 기능

- 사용자가 마음에 드는 레시피를 좋아요하거나 스크랩할 수 있고 별점과 한줄평을 남길 수 있는 레시피 리뷰 기능 제공
- 레시피에 대한 리뷰를 작성하고 다른 사용자와 상호작용할 수 있는 서비스 제공

한계

1. 오프라인 사용 및 접근성

웹 서비스는 주로 인터넷에 연결된 디바이스를 통해 이용되므로 오프라인 상황에서는 이용이 어려울 수 있습니다. 특히 모바일에서의 접근성이 떨어질 수 있습니다.

2. UI/UX 측면의 사용자 경험 한계

웹을 통해 본 서비스를 사용하는 사용자는 네이티브 앱 서비스에 UI/UX 측면에서 사용자 경험이 떨어진다고 생각할 수 있습니다.

3. 기능 제한

웹에서 제공되는 기능이 모바일 네이티브 앱에 비해 제한될 수 있습니다. 예를 들어, 푸시 알림, 카메라 사용등이 웹에서는 구현이 어려울 수 있습니다.

6. 기술적 요구 사항

▼ Frontend

- IDE : VScode
- React, Typescript
- Sass
 - css 작성을 효율적으로 하기 위해 사용

- React Router
 - 페이지를 새로고침 하지 않고도 다양한 뷰를 렌더링 하기 위해 사용
- Redux
 - 상태 관리를 위한 라이브러리 사용
- Vercel
 - 페이지 배포를 위해 사용
- 반응형 디자인
 -

▼ Backend

- IDE : IntelliJ, VScode
- JDK
 - Java 17을 실행할 수 있는 JDK
- Spring boot 3.1.1
- Spring Data JPA & Querydsl
 - DB와 프레임워크를 연결하는 역할
 - 간단한 쿼리는 JPA를 통해 작성할 수 있지만 상황에 따라 쿼리를 사용하기 위해 QueryDSL을 같이 사용을 위해 채택
- Spring Security
 - 로그인 및 보안 관련 사항을 위해 사용
- Postman
 - 개발한 API를 테스트하고, 테스트 결과를 공유하여 API 개발의 생산성을 높여 주는 플랫폼
 - 작성된 Rest API를 테스트 하기위해 사용
- MySQL
 - 무료 오픈 소스, 시장 점유율 상위, AWS RDS 연동 가능과 같은 이유로 채택
- RestDoc
 - Controller Test code 작성된 결과를 API 문서로 만들어서 사용하기 위해 채택

- Redis
 - Refresh 토큰의 사용을 위해 채택

▼ Infra

- AWS : EC2, S3, RDS
 - spring, react로 구성된 프로젝트, jenkins를 위한 EC2 서버
 - mysql을 위한 RDS
 - 이미지 저장을 위한 S3
- Nginx
 - 웹 서버 로드 밸런서 및 HTTPS 구현시 리버스 프록시 담당
- docker
 - 서버 제작 과정에서 배포 편의성 및 확장성을 위해 사용
- Jenkins
 - 빌드, 테스트, 배포 프로세스를 자동화
- JUnit
 - 단위 테스트 프레임워크이다.
 - 단위테스트를 통해 구성한 메서드와 로직이 원하는 결과를 출력하는지 확인하는 역할

7. 예산 및 자원

프로젝트 진행 시 예산 및 자원 비용은 크게 클라우드 서버 및 도메인 네임 등록 비용이다.

클라우드 서버의 경우의 경우 다음과 같이 서버를 구성할 것 같다. 각각의 서버의 경우 spring, react로 구성된 프로젝트, jenkins를 위한 ec2 3대, mysql을 위한 rdb, 이미지 저장을 위한 s3, 그리고 리프레시 토큰의 사용을 위한 redis를 사용하여 클라우드 서버를 구성할 예정이다.

이때 사용 비용은 ec2의 경우 배포에 사용할 정도의 성능이면 서버당 ncp 기준으로 34000원이다. Rdb,redis 최소 비용 5000원이 측정이 되었다. 그렇기에 비용적 문제로 인해 초반에는 docker 컨테이너를 사용하여 개발자 기준 통합 테스트 환경을 설정할 예정이다. 배포의 경우 6월에 프로토 타입 및 추가 기능이 구현이 되면 진행할 예정이다. 사용자 테스트의 경우 해당 docker 환경에서 구현된 웹을 사용하여 테스트를 진행하고 지인 위주의 피드백을 받을 예정이며, 6월에 배포 후에는 이제 익명의 참여자들을 통해 테스트를 받을 예정이다.

다만 이미지의 저장의 s3의 경우 초반 기능 구현에 필요한 상황이다보니 기존의 인원 ncp 크레딧을 통해 사용할 예정이며, 테스트를 위해 사용하는 것이기에 이미지의 최대 크기를 설정하여 사용할 예정이다.

DNS의 경우 저렴한 도메인을 사용하면 최소 비용으로 이용이 가능한 상황이다. 그렇기에 서버에 dns를 연결해서 사용할 예정이면 구매해서 연결할 예정이다.

8. 예상 결과

프로젝트

프로젝트의 목표 달성이다. 동네 기준의 요리 플랫폼을 완성하여 사용자들에게 음식과 관련된 소통 서비스 플랫폼을 구현하여 사람들이 음식에 대한 접근 장벽과 소통에 대한 장벽을 낮출 수 있을 것이라 생각한다. 그렇기에 프로젝트 서비스를 통한 결과는 크게 세 가지라고 생각한다.

1. 음식에 대한 장벽의 간소화

기존의 여러 어려움(시간, 재료, 조리 노하우, 메뉴 선택)등의 이유로 사람들이 음식을 만들어 먹는 것에 대해 벽이 있다고 생각한다. 다만 해당 서비스를 통해 본인이 가지고 있는 재료 내에서, 본인이 투자할 수 있는 시간만큼, 본인이 조리할 수 있는 난이도, 본인이 관심있는 메뉴를 통해 직접 만든다는 장벽을 낮추는 효과를 기대한다.

2. 사람간의 소통

시간이 지나면서 SNS가 발달하고 여러 소통수단이 생겼지만 동네 사람들과의 주변 사람들 간의 소통의 창구는 점점 줄어들고 있다고 생각하였다. 소통의 창구는 줄어들었지만 그래도 주변의 사람들과 소통을 원하는 사람들은 존재한다고 생각한다. 기존에 존재하는 플랫폼인 당근, 번개장터, 네이버 동네 모임등과 같은 경우가 이러한 상황을 반영하고 있다고 생각한다. 이 때 요리의 주제로 한 플랫폼은 여러 존재하였지만 요리를 기준으로 소통을 하는 지역 플랫폼은 없다고 파악을 하여 진행한 프로젝트이다. 해당 프로젝트를 통해 사람이 살기 위해 필수적인 음식이란 요소를 주제로 하여 사람들이 소통을 하며 공동체를 만들며 나아간다는 결과를 예상했다.

3. 경제적 효과

혼자서 사기에는 부담스러운 식자재, 혹은 일정 금액을 사면 쓸 수있는 쿠폰, 내가 필요한 양을 넘어가는 식자재 등의 여러가지의 소비에 관한 상황들이 있다. 이때 내가 산, 사야할 식자재를 누군가와 같이 사고, 가격에 대한 부담을 덜어내는 상황을 해당 서비스를 통해 만들어 낼 수 있다고 생각한다.

프로젝트를 통해 진행한 서비스를 통해서 음식에 대한 장벽 간소화, 소통, 경제적 효과를 해당 서비스의 핵심 기대 결과라고 생각한다.

9. 일정 계획 - 마일스톤

- 매주 화, 목, 토 기술 및 진행사항 논의
- 로컬 테스트 환경 구성 및 배포 환경 구성 후 새로운 기능 생성 시 반영하여 테스트 진행하기
- 각 기능 개발 시 단위 테스트 및 개발 테스트 진행
 - cicd 자동 테스트를 통해 통과가 되면 build 및 배포 진행

240327 ~ 240330

- 시스템 설계 및 기능 및 API 정리
- 공공데이터를 통한 레시피 데이터 획득 및 재가공

240331 ~ 240403

- CORS 및 Security등의 Frontend 및 Backend 연결
- 기본 기능 구현을 위한 객체 및 테이블 연결
- Docker file 및 compose를 통한 로컬 테스트 환경 구현

240404 ~ 240406

- 사용자, 레시피 기준으로 중요도(1~2)의 기능 구현하기
 - 승은 : frontend 및 UIUX 구현
 - 민수 : 사용자 기능(회원가입, 로그인 로직, 회원 조회) 구현 및 사용자 및 레시피 단위 테스트 작성
 - 세현 : 레시피 기능 (레시피 생성, 조회, 전체 조회, 검색) 기능 구현

240407 ~ 240410

- 웹소켓 사용을 통한 채팅 기능 구현하기
- 각 기능별 RestDocs를 통한 API 문서작업 진행

240411 ~ 240413

- 채팅 기능까지 포함한 로컬 테스트 환경에서 시험 테스트 진행
- 이미지를 위한 S3 설정 및 기능 추가

240414 ~ 240420

- 회원 휴대폰 인증 기능 구현 (NCP - SENS API)
- 중요도 3인 각 부분의 기능 구현 (아이디, 비밀번호 찾기 및 수정, 사용자 삭제)

240421 ~ 240427

- 각 기능별 추가 기능 탐색 및 API 정리
- 기능별 API의 중요도 재설정

240428 ~ 240504

- AWS 및 NCP를 통한 서비스 배포 시도 및 진행
- 실제 웹 환경에서의 테스트 진행 및 시도

240505 ~ 240511

- CI/CD 파이프라인 설정 및 테스트
- 기능별 단위 테스트 작성 및 API 문서화
- frontend : 기능 추가된 화면을 기준으로 UI/UX 리뷰 및 수정 부분 조회

240512 ~ 240518

- 검색기능 고도화
 - 기존의 음식이름 검색에서 재료 및 사용자 검색까지 가능하도록 검색 기능 고도화
 - 검색 기능 고도화를 위해 채택해야하는 기술 고려를 위한 논의

240519 ~ 240525

- 프로젝트 리팩토링 사항 찾기 및 정리 (중요도 및 우선도 정리)

240526 ~ 240601

- 프로젝트 리팩토링 실시

240602 ~ 240608

- 캡스톤 프로젝트 마무리 및 추후 진행 계획 작성

