

프로젝트 완료 보고서

프로젝트 개요

- 팀을 구성하여 애플리케이션의 기획부터 디자인, 구현까지 함으로써 원활한 의사소통 능력과 리액트 기술을 증진시킨다.

기획의도 & 목표

- 애플리케이션 구현에 필요한 리액트 기술을 학습하고 활용하며 실력을 증진시킬 수 있도록 함
- 정해진 기간 안에 완성도 있는 애플리케이션을 구현하여 완성의 경험과 자신감을 기를
- 팀원들 간의 원활하고 적극적인 의사소통을 통해 프로젝트를 더 나은 방향으로 나아갈 수 있게 함

각 담당 업무

이성수	리더, 기능 구현 담당
성영은	프로젝트 진행 상황 기록, 기능 구현 담당
정하솔	github 관리, 디자인 총괄 담당
조유빈	디렉토리 관리, 기능과 디자인 담당

요구사항 분석

- 기능적 요구사항

서비스	필요 기능
회원가입	절차에 맞게 회원가입을 할 수 있도록 input 제공
로그인	알맞는 아이디와 패스워드를 입력하여 로그인
로그아웃	로그인 후 해당 계정 로그아웃
아이디 중복 체크	다른 사용자와 아이디가 중복되는 상황을 방지
냉장고 등록	사용자가 사용할 냉장고 등록
식료품 등록	식료품 목록에 사용자가 가지고 있는 식료품 추가
식료품 추가	식료품의 이름, 수량, 보관위치, 소비기한, 메모 등을 입력할 수 있게 구성

서비스	필요 기능
식료품 삭제	등록했던 식료품을 삭제하는 기능
식료품 수정	등록한 식료품에 대한 정보 수정이 필요할 시에 정보를 수정하는 기능
식료품 검색	원하는 식료품을 검색 기능을 통해 쉽게 찾을 수 있음
목록 정렬	사용자가 원하는 방식으로 목록을 확인할 수 있도록 다양한 정렬 기능 제공
보관 위치별 분류	실온, 냉장, 냉동 등 식료품의 보관 위치를 기준으로 하여 분류하고 볼 수 있도록 함
날짜별 식료품 확인	식료품의 구매일자를 기준으로 달력 기능을 사용하여 한눈에 확인할 수 있게 함
외부 쇼핑몰 링크 연결	외부의 온라인 쇼핑몰로 연결되는 배너를 배치하여 식료품의 정보를 확인하거나 주문할 수 있도록 함
장보기 목록 작성	사용자가 자유롭게 등록/삭제 가능한 장 보기 목록 기능 제공

• 비기능적 요구사항

보안 및 개인정보 보호	로그인 및 사용자 정보 처리 시, 개인정보를 안전하게 보호함
사용자 친화성	직관적이고 사용자 친화적인 인터페이스를 제공하여 쉽게 기능을 이용할 수 있도록 함
심미성	보기 좋은 디자인과 적절한 레이아웃 배치를 통하여 사용자가 이용 시 시각적인 편안함을 느낄 수 있게 함

개발 환경

개발 Tool	vscode, Adobe Illustrator
관리 Tool	Github, Figma, Notion, Google spread sheet
앱 구현	REACT, CSS
REACT 플러그인 / 라이브러리	@fullcalendar, react-router-dom, uuid

작업 순서

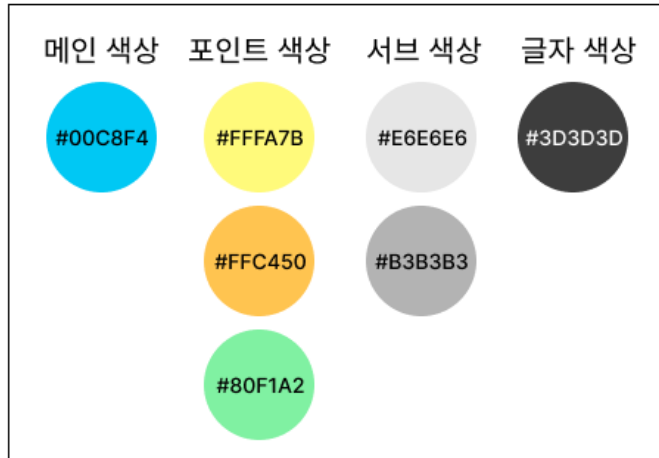
- 사전 기획
 - 개발할 애플리케이션의 기능 선정
 - 팀 내 역할 분담과 팀명 선정
 - 사용자 시나리오 작성
 - 유스케이스&워크플로우 작성

- 레퍼런스 어플 선정 후 분석
- 디렉토리 구조 작성
- 코딩 및 디자인 가이드 작성
- 개발 작업
 - 프로젝트 설정: React 설치 후 필요한 플러그인/라이브러리 추가 설치
 - 디렉토리 생성
 - 페이지 레이아웃 작업
 - 각 페이지에 필요한 기능 구현
 - localStorage를 통해 데이터를 저장한 후 페이지에 배치
 - React Router를 사용하여 페이지 간 원활한 이동이 가능하게 함
 - full-calendar를 사용하여 달력의 형태를 만들고 해당하는 날짜에 아이템이 이벤트도 뜨도록 함
- 디자인 작업
 - UI 프로토타입 작업
 - 메인 색상과 포인트 색상, 사용될 폰트 선정
 - 애플리케이션에 삽입될 일러스트, 아이콘 작업
 - 전체적인 분위기를 통합하여 UI디자인 완성 후 스타일 작업
- 그 외 프로젝트 작업
 - 발표 준비 후 발표
 - 피드백을 바탕으로 한 프로젝트 수정
 - 프로젝트 완료 보고서 작성

가이드 라인

- 디자인 가이드

색상



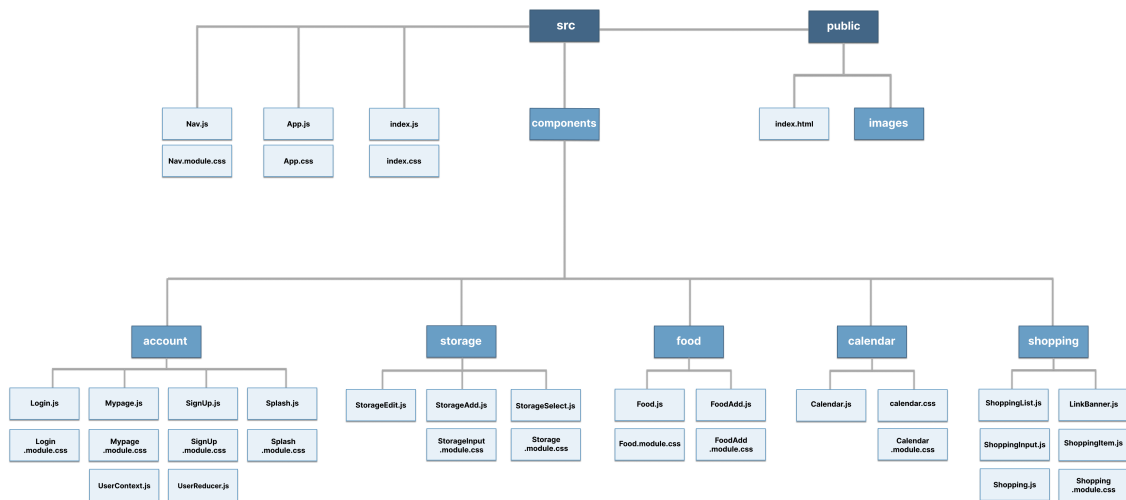
폰트



- 코딩 가이드
 - 클래스 네임은 쉽게 알아볼 수 있고 누구나 쉽게 납득 가능한 이름으로 가능한 이름으로 작성한다. 네 단어 이상의 사용을 지양하고 카멜 표기법으로 작성한다.
 - const 선언방식을 사용한다.
 - 따옴표는 큰 따옴표를 사용한다.
 - 이벤트 핸들러 함수를 사용한다.
 - Prettier를 사용하여 저장한다.
 - 설명이 필요한 부분은 주석을 통해 작성한다.
 - 화살표 함수를 사용하여 작성한다.
 - 들여쓰기는 space2로 통일한다.
 - 특별한 기능이 없는 곳에 사용하는 태그는 <div>로 통일한다.
- REACT 작업 시 설치할 라이브러리/플러그인
 - fullcalendar
 - uuid
 - react-router-dom

프로젝트 내용

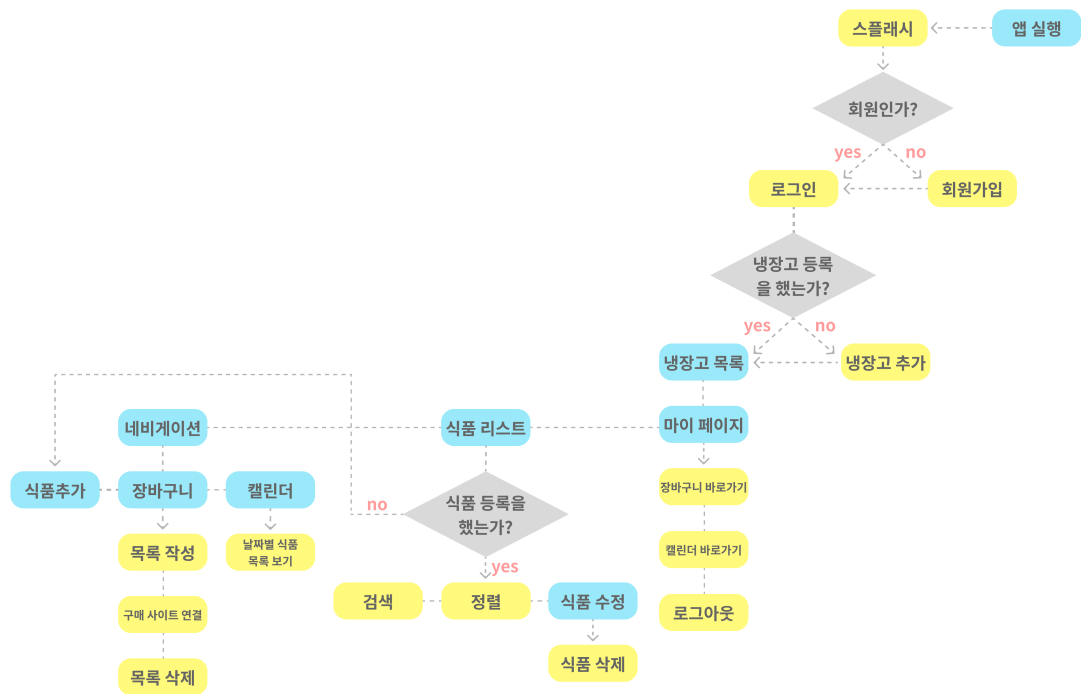
- 디렉토리 구조



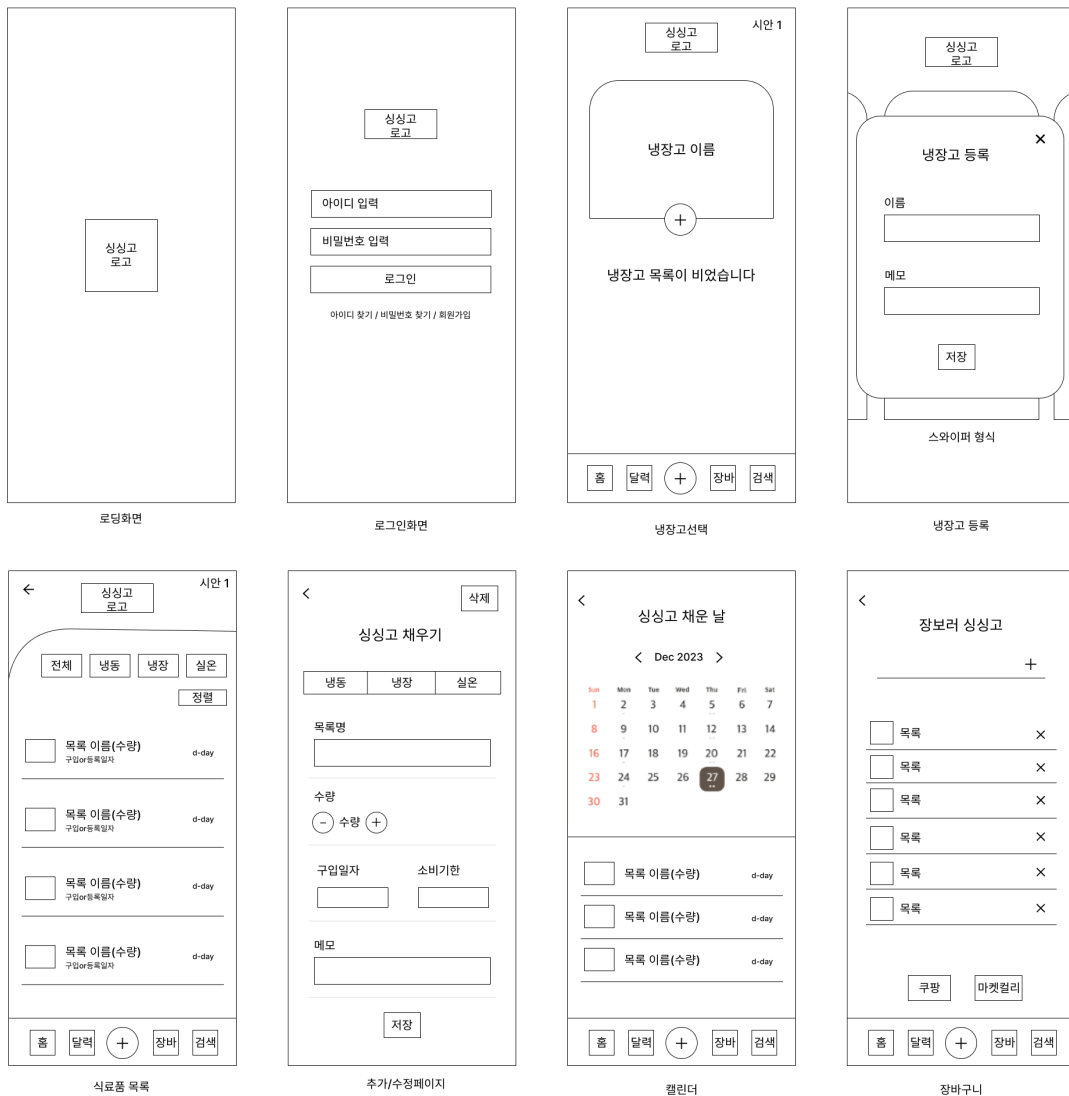
• 사용자 시나리오

	실행 전	실행 중	실행 후
사용자 행동	<ul style="list-style-type: none"> • 배가 고파 냉장고 문을 연다. • 냉장고 문을 여니 소비기한이 지난 식품들과 썩어가는 채소들이 보인다. • 냉장고 정리에 대한 필요성을 느껴 관련된 어플을 사용하기로 한다. 	<ul style="list-style-type: none"> • 냉장고에 있던 상한 음식을 버리고 새로운 식료품과 기존 식료품을 정리하여 소비기한 및 보관 날짜를 어플에 기록한다. • 어플을 통해 어떤 재료가 있는지 확인한 후 장보기 목록을 작성한다. • 요리 시 소비기한이 임박한 재료들을 우선 사용하여 요리를 한다. 	<ul style="list-style-type: none"> • 어플을 통해 냉장고 속 재료를 파악하기 쉬워져 장을 보기에 훨씬 수월해졌다. • 가지고 있는 재료를 한눈에 파악할 수 있어 요리할 메뉴를 정하는 데에도 도움이 되었다.
사용자 니즈	<ul style="list-style-type: none"> • 냉장고 정리를 위한 어플을 찾는다. 	<ul style="list-style-type: none"> • 내가 가지고 있는 식료품을 확인하고 싶다. • 내가 구매한 식료품의 소비기한을 한눈에 확인하고 싶다. • 사야할 식료품을 기록할 수 있었으면 좋겠다. • 소비기한이 적게 남은 순서대로 식료품을 볼 수 있었으면 좋겠다. • 내가 어떤 날에 어떤 식료품을 구매했는지 한눈에 확인하고 싶다. 	
기능	<ul style="list-style-type: none"> • 사용자가 필요로 하는 어플을 쉽게 찾을 수 있도록 어플을 디자인하고 소개를 함 	<ul style="list-style-type: none"> • 식료품의 구매날짜와 소비기한 추가 • 냉장/생동/실온 카테고리를 구분하여 식료품 확인 • 장 볼 목록을 작성할 수 있는 페이지 구성 • 사용자가 원하는 기준으로 목록을 정렬 가능 • 달력을 통해 구매한 식료품을 확인 가능 	

• 워크플로우



- UI 프로토타입



- 페이지 구성
 - 메인 스플래시
 - 로그인 및 회원가입
 - 마이페이지(계정 관리)
 - 냉장고 등록 및 관리
 - 식료품 목록 추가 및 수정
 - 장바구니 목록
 - 달력(날짜 별 구매품 확인)

주요 기능

- 로그인 및 회원가입
 - 아이디 중복 확인 기능을 추가하여 사용자의 아이디가 중복되는 것을 방지함
 - 비밀번호 등록 시 두 번 입력하게 하여 사용자의 실수 및 혼란을 방지함
 - 아이디와 비밀번호는 설정해 놓은 기준을 충족해야만 등록 가능하게 하여 관리 및 보안에 용이하게 함
 - 마이페이지에서 로그인한 계정의 정보를 확인할 수 있고 로그아웃 가능
- 냉장고 등록
 - 냉장고 이름과 메모를 작성할 수 있도록 함
 - 수정 기능을 통해 사용자가 원하는대로 정보를 수정할 수 있음
- 식재료 목록
 - 검색 기능을 통해 식료품의 수량이 많을 때에도 편리하게 원하는 식료품을 찾아낼 수 있음
 - 카테고리별로 식료품을 확인할 수 있게 함
 - 가나다순, 구매일자순, 소비기한순으로 정렬기준을 세우고 제공
 - 남은 소비기간을 한눈에 확인할 수 있도록 제공
- 식재료 추가
 - 카테고리, 식료품명, 수량, 구매일자, 소비기한, 메모 등을 간단히 작성할 수 있도록 제공
 - 등록 후 수정 기능을 통해 수정 및 삭제 가능

문제 및 해결

문제점	해결 방법
보관 중인 식재료 목록 - 새로고침을 하면 등록했던 목록들이 사라지는 문제	localStorage를 이용하여 브라우저에 데이터를 저장 후 불러오는 방식으로 유지
디자인 - 색상 대비가 약하고 통일감이 없다는 피드백	기존에 선정했던 색상에서 채도를 높이고 일러스트와 아이콘들을 컨셉에 맞게 수정
달력 기능 - 이벤트를 적용하는 달력의 구현을 어떤 식으로 진행해야 할지 갈피를 잡지 못함	fullcalendar 라이브러리를 발견한 후 달력 기능과 날짜별로 이벤트를 띄우는 기능을 활용하여 원하던 기능 구현

문제점	해결 방법
식료품을 선택한 후 삭제 버튼을 누르면 선택한 식료품이 삭제되어야 하는데 랜덤의 식료품이 삭제되는 문제	uuid를 이용하여 식료품 아이템마다 고유의 id값을 가지게 한 후 삭제 함수에서 선택한 아이템의 id를 찾아 삭제할 수 있도록 코드 수정

프로젝트 완료 리뷰

- 아쉬운 부분
 - 처음에 기획했던 모든 기능을 구현하지 못했던 점이 아쉽다.
 - mongoDB를 배운 후 발표까지의 시간이 짧아 데이터 관리에 mongoDB를 직접 사용해볼 수 없었던 점이 아쉽다.
 - 발표 시 발표의 목적을 잘 파악하지 못해 개발 작업에 대한 설명이 부족했던 점이 아쉽다.
- 잘한 부분
 - 팀원들 간의 의사소통이 원활히 잘 이루어져 피드백과 수정이 금방 이루어졌다.
 - 의도한 느낌대로 디자인과 레이아웃의 배치가 잘 이루어졌다.
 - 역할 분배가 잘 이루어져 각자의 작업에 더욱 집중할 수 있었고, 그로 인해 좋은 결과물을 낼 수 있었다.
 - 어렵고 막막했던 기술들을 팀원들 모두 도전해보고 결과물을 만들어 냈다.
 - 팀원들 모두가 프로젝트에 적극적으로 참여하고 작업을 구상해냈다.
- 배운 점 및 주관적인 의견
 - 기획부터 구성, 디자인, 구현까지 모두 해보는 경험 자체가 즐거웠고, 과정은 힘들었지만 만족할만한 결과물을 만들어내어 자신감을 얻게 되었다.
 - 달력 기능을 구현하며 다양한 라이브러리 및 플러그인을 찾아보게 되었고, 기능에 따른 캘린더 관련 라이브러리에 대해 많이 알게 되었다.
 - localStorage에 대한 개념을 제대로 알게 되었고 곧바로 활용까지 하여 잘 이해할 수 있었다.
 - 팀원들 간 의사소통이 팀 프로젝트에서는 정말 중요한 점이라는 걸 깨달을 수 있었다. 원활히 의사소통이 이루어졌기 때문에 만족스러운 애플리케이션을 구현할 수 있었다고 생각한다.
 - 처음 계획했던 기능들을 모두 구현하고 싶었지만 능력 부족, 시간 부족 등의 이유로 전부 구현하지 못했다. 계획할 때 주어진 시간과 각자의 능력을 고려하여 계획을 했으면 더 효율적으로 시간을 사용할 수 있었을 것 같다.

- 전반적으로 정말 만족스러운 프로젝트였다. 성공 경험을 통해 자신감과 흥미를 얻을 수 있었고 더욱 많은 기술을 알아가고 싶다는 생각을 하게 된 프로젝트였다.