

< 리액트 프로젝트 완료보고서 >

- 신선하조의 싱싱고 -

프로젝트 개요

- 싱싱고는 React를 응용하여 냉장고 속 식재료의 관리에 도움을 주는 Web App

기획의도 & 목표

- 수업을 통해 학습한 REACT를 응용해 코딩 실력 향상
- 정해진 기간 내에 완성도 있는 UI 디자인
- 피드백을 주고 받으며 더 나은 방향의 냉장고 관리 어플 '싱싱고' 완성

개발 환경

- 프로젝트 관리 : GitHub / Google Drive / Notion
- 개발 프로그램 : VsCode / Adobe Illustrator
- Front : REACT / .module.css / JS FullCalendar 플러그인 / react-router-dom
- 언어: JavaScript
- 배포 : GitHub-Pages

담당 업무

- 이성수 : 리더 / 기능 구현
- 성영은 : 발표 & 일정 기록 / 기능 구현 / CSS 작업
- 정하솔 : GitHub 관리 / 디자인 총괄 / CSS 작업 / PPT 제작
- 조유빈 : 디렉토리 관리 / 기능 구현 / CSS 작업 정리

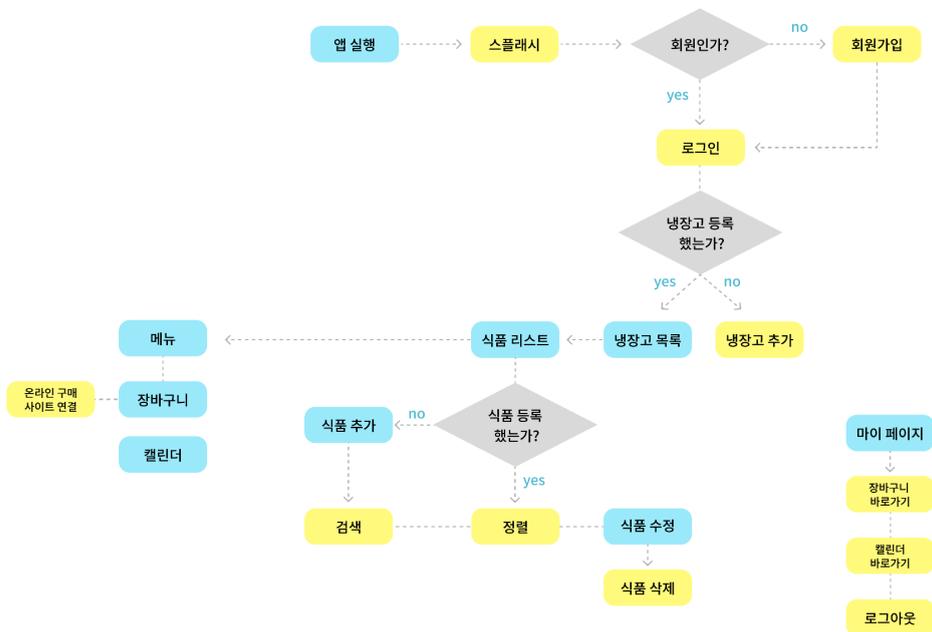
작업 일정

12/01	구현할 어플 기획 아이디어 선정
12/02 - 04	사용자 시나리오 및 워크플로우 작성하여 어플의 기능 요소 흐름 파악
12/04 -05	워크플로우를 기반으로 기능적,비기능적 요구사항 분석
12/05-06	페이지 구성에 맞게 프로토타입 작업 -> 디렉토리 구조를 작성하여 개발 폴더 구조 파악
12/07	기능 부분과 UI 디자인에 참고할 레퍼런스 어플 서치 / 간단한 구조 작업

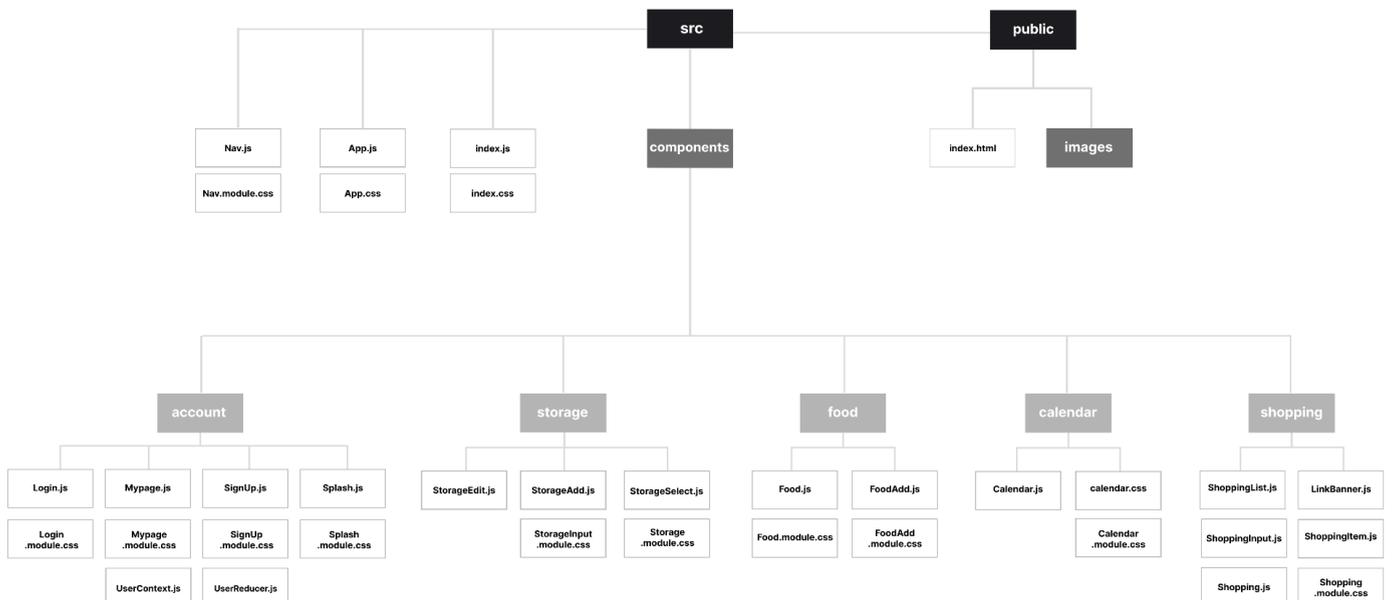
12/07-13	본격적인 기능 구현 / 디자인 시안 작업 및 완성 (중간 중간 피드백)
12/14-20	기능 오류 해결하면서 CSS 스타일 작업 수정 반복하며 완성
12/20	GitHub Page로 배포 후 생기는 오류 해결 / PPT 제작과 발표 준비
12/21	발표
12/21-26	추가 작업 (mongoDB로 데이터 관리)

프로젝트 내용

• 워크플로우



• 디렉토리 구조



- 요구사항 분석

기능적 요구사항	
스플래시 화면	어플 실행 했을 때 보이는 첫 화면
로그인	고유한 아이디와 비밀번호로 로그인
회원가입	아이디 중복 체크 / 비밀번호 조합 요구하려 보안성 높임
로그아웃	로그인 상태에서 로그아웃
마이페이지	사용자의 아이디 정보를 볼 수 있고, 페이지 내에 로그아웃 기능
냉장고 목록	여러 대의 냉장고를 등록해 따로 관리 할 수 있도록 함
식재료 목록	냉장고 속 식재료의 소비기한과 구매날짜, 수량 등을 목록으로 관리
냉장고 / 식재료 수정	잘못 등록했거나 식재료를 사용했을 때 수정 가능
냉장고 / 식재료 삭제	기록할 필요가 없어졌을 때 삭제할 수 있는 기능
식재료 검색	원하는 식료품을 쉽고 빠르게 찾을 수 있도록 검색
식재료 목록 정렬	사용자가 원하는 방식으로 정렬(구매날짜 / 소비기한 / 이름 순)
식재료 카테고리	보관위치에 따라 분류하여 확인 가능(냉동 / 냉장 / 실온)
장보기 목록	장 봐야할 목록을 미리 작성해서 확인할 수 있음
온라인 구매(외부링크)	장 보러 나가기가 귀찮을 때 온라인 구매 사이트를 외부링크로 연결
캘린더	식재료를 구매한 날짜를 캘린더로 확인 가능
비기능적 요구사항	
보안	로그인으로 보안 프로토콜을 적용하여 개인정보를 안전하게 보호
사용자 친화성	직관적인 UI를 제공해 쉽게 기능을 이용할 수 있도록 함
	오류시 오류페이지를 작성하여 사용자가 상태를 인식할 수 있도록 제공
확장성	새로운 기능 및 업데이트를 효율적으로 추가할 수 있도록 시스템의 확장성을 고려함
품질 테스트	지속적인 테스트로 시스템의 품질을 확보하고 오류를 최소화함

프론트엔드 개발 작업 순서

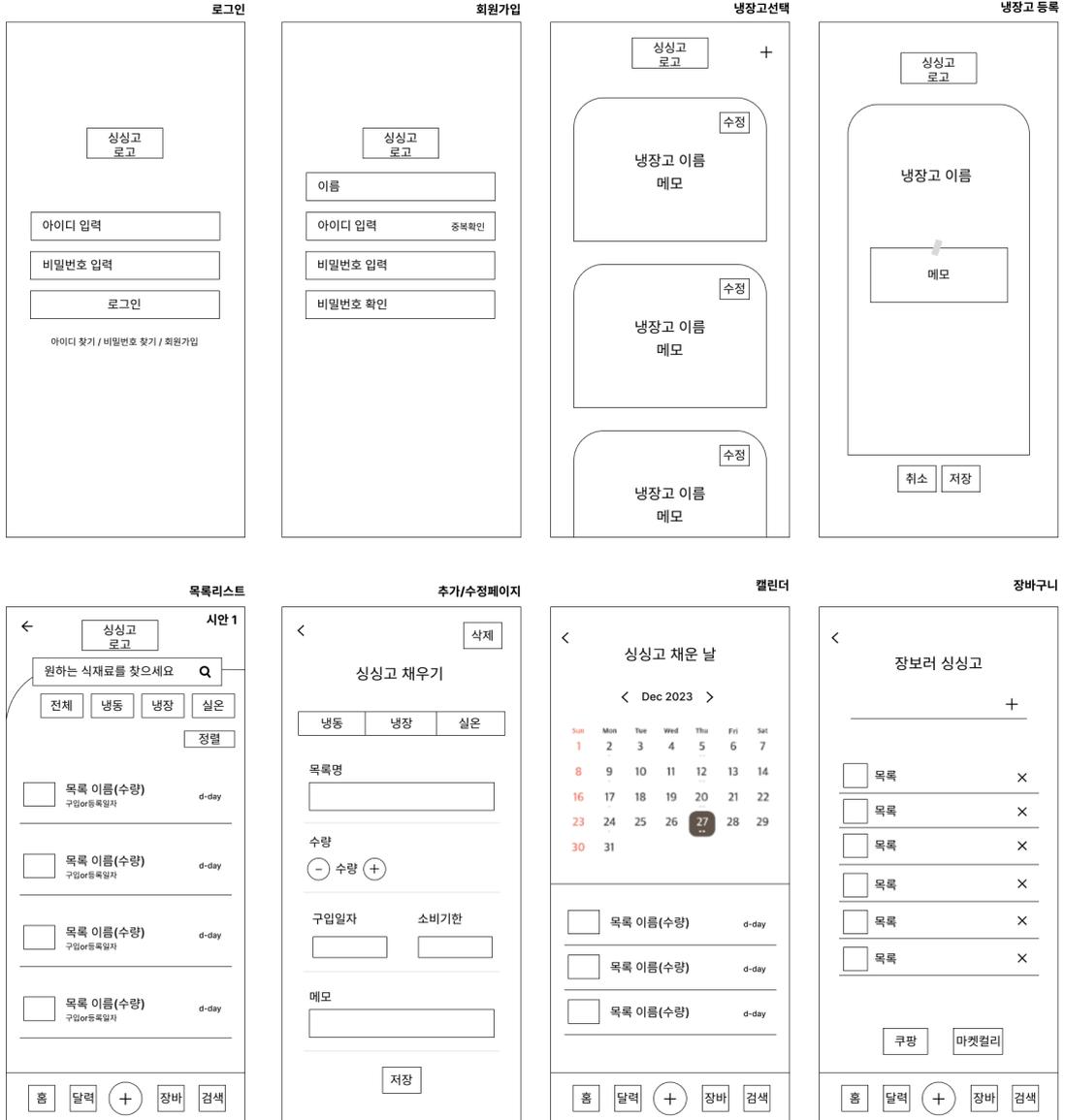
- 코딩 가이드 작성

코딩 가이드	
className	최대 세 단어 이상만 사용하고 카멜 표기법으로 작성
변수 선언	const 사용
따옴표	“큰 따옴표”를 사용
이벤트	이벤트 핸들러 함수 사용
부가 설명	주석을 사용해 작성
함수	화살표 함수로 작성
들여쓰기	space 2로 통일
영역 태그	특별한 경우를 제외하고 <div>로 통일
코드 정렬	VsCode 내에 Prettier를 사용

- 작업할 REACT 폴더 생성(디렉토리 구조)
- 각 폴더 안에 컴포넌트 작성
- 필요한 패키지 설치
 - react-router-dom/uuid/JS FullCalendar/gh-pages
- 라우팅 설정
 - React Router로 페이지 간의 이동 관리
- 상태 관리
 - React Hook 중 useState와 useEffect로 동적인 값을 관리
- React 앱 빌드
 - 배포하기 전에 빌드해서 정적인 파일 생성
- 테스트 및 배포
 - GitHub-Pages로 테스트와 배포

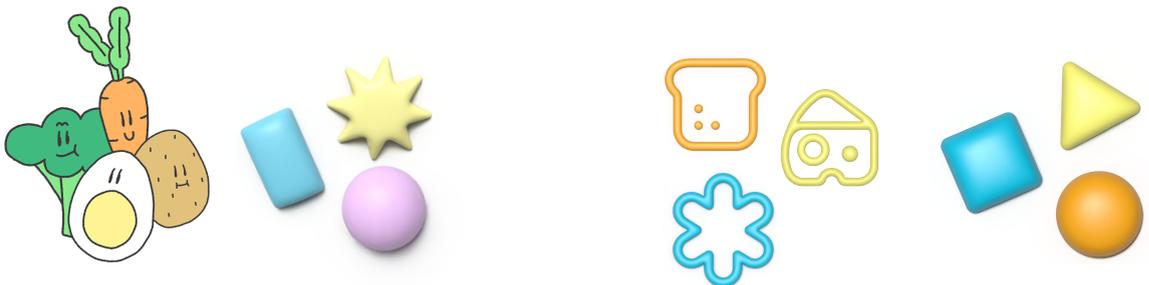
UI 디자인 작업 순서

- 프로토타입 제작하여 레이아웃 잡음

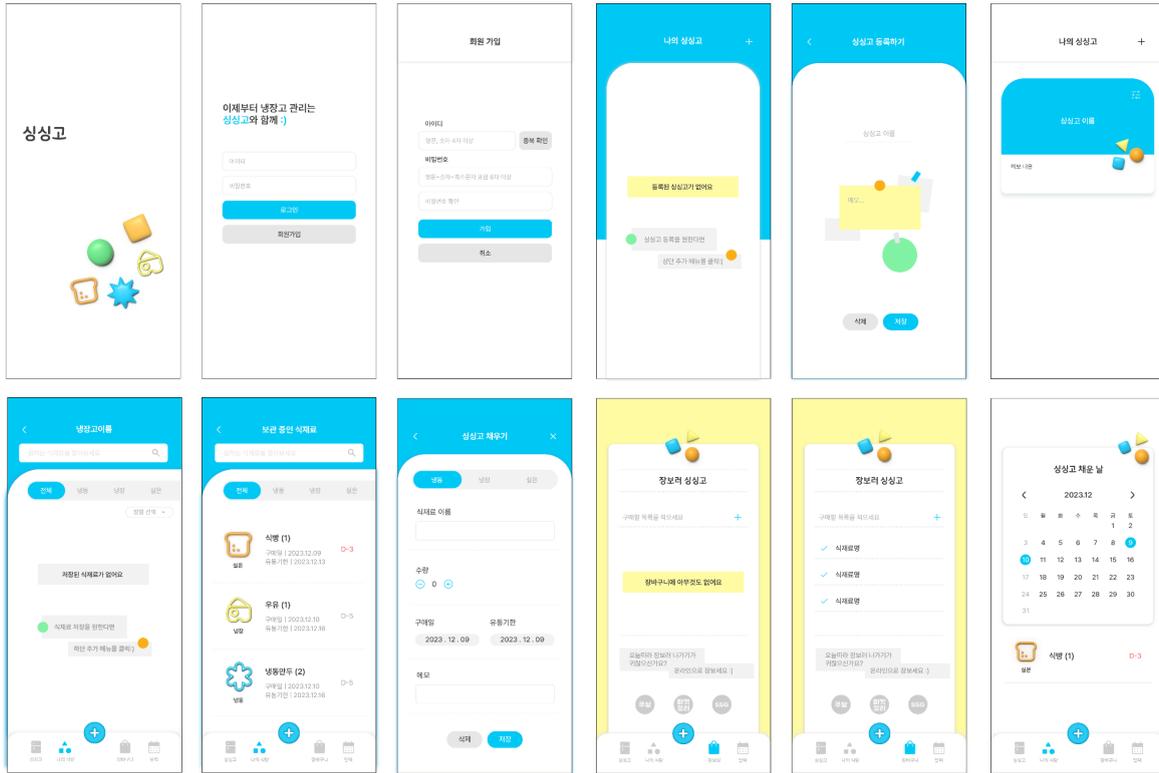


- 디자인 컨셉에 맞는 일러스트 제작(+수정)

- 수정 전 → 후



● 시안 작업



● 스타일 가이드 작성

스타일 가이드	
본문 폰트	font-family: 'Pretendard-Regular'; (웹폰트로 연결)
포인트 폰트	font-family: 'SOYOMapleBoldTTF'; (웹폰트로 연결)
본문 폰트 사이즈	1.1rem
제목 폰트 사이즈	1.3rem
작은 폰트 사이즈	0.9rem
작업 사이즈	380*844(아이폰 12 프로 기준)

메인 #00C8F4	포인트1 #FFFBA3	포인트2 #FBAE17	포인트3 #6AE589	그레이1 #E6E6E6	그레이1 ##B3B3B3	폰트 #6AE589
---------------	-----------------	-----------------	-----------------	-----------------	------------------	---------------

문제 및 해결

문제	해결
보관 중인 식재료 목록을 새로고침 하면 등록했던 데이터가 초기화 됨	로컬스토리지에 저장해서 데이터 저장
초기 디자인에서 사용한 색감의 대비가 약하고, 전체적으로 통일성이 없어 컨셉이 명확하지 않았다는 피드백을 받음	메인 컬러의 채도를 높이고, 기존 작업했던 일러스트를 빼고 마그넷 컨셉으로 통일
table로 캘린더를 직접 만들어서 저장된 데이터를 가져오려고 시도 했으나 실패	JS FullCalendar 플러그인을 활용해 대체
식재료 목록에서 식재료 삭제 버튼을 누르면 목록이 랜덤으로 삭제됨	uuid로 고유의 id값을 주어 삭제 이벤트 함수에서 선택한 아이템의 id를 찾아 실행함
소비기한 디데이 임박 시 폰트 색상이 변경되는 기능을 if 문으로 구현하려 했지만 실패	css 우선순위로 확인해 보고, console을 통해 받아오는 데이터도 확인했지만 시간 안에 해결하지 못함. 추후 보완 예정
GitHub-Pages에 테스트 배포 시 이미지가 안 뜨는 문제 발생	images 폴더가 절대 경로에 있었으나 src 경로 작성 시 /앞에 붙어있던 .을 빼서 해결

완료 리뷰

- **잘한 점**
 - 팀원들의 역할 부담이 적절하게 이뤄지면서 서로의 부족한 부분을 채워주면서 시간 안에 완성도 있는 결과물을 만들 수 있었던 것 같음
 - 진행 과정에서 막히거나 오류가 발생했을 때 서로 의견을 나누며 같이 해결한 점
 - 계획했던 기능의 대부분을 구현하고, 안정감있는 UI 디자인을 한 점
- **아쉬운 점**
 - 기능의 대부분을 구현하기는 했지만 데이터 저장을 로컬로 대체하면서 데이터 관리가 어려워진 것 같아 아쉬움이 남음
 - 식재료 목록에 주요 기능이 몰려 있어 구현 당시 오류가 많았는데, 특히 수정 페이지 오류를 해결하기 위해 업데이트에 유리하도록 부분별로 나뉘던 컴포넌트를 합치게 되었는데 그 점이 아쉬움
 - 캘린더 플러그인을 사용하면서 플러그인의 기본 스타일 성격이 있어 CSS 수정에 한계가 있던 점

- **배운 점과 의견**

- 기존에 있던 어플을 참고하기는 했지만 그럼에도 무에서 유를 창조하는 느낌이라서 시작은 너무 막막했음. 그러나 조금씩 틀이 잡히고, 아웃풋이 만들어지다보니 우리도 할 수 있구나를 느낌.
- 퍼블리싱 작업 시까지 고려해서 UI 디자인 작업을 하다보니 디자인 과정은 머리가 너무 아팠지만, 퍼블리싱을 고려하면서 디자인 작업을 할 수 있게 되었다는 점이 뿌듯했음
- 무언가를 기획하는 것을 어려워하는 편인데, 이번 프로젝트를 진행하면서 생각을 넓혀가는 방법을 조금은 터득한 것같아 뿌듯함
- 각자 잘 할 수 있는 분야에 맞춰서 담당을 나눴는데, 그 밸런스 좋았다고 생각함. 그리고 밸런스가 좋으면 개개인의 역량이 아무리 부족 하더라도 채워줄 수 있는 누군가가 있어 뭐든 할 수 있음을 깨달음