



Cookology

Food Recommendation Project

TEAM : OAOC
One Accepted One Chicken



Index



1. 팀원소개

개인 프로필
프로젝트 내 역할



2. 기획의도

주제 선정 이유
유사사이트 장단점 분석



3. 사용기술



4. 기능정의서

Index



5. 개발일정



6. Usecase Diagram



7. 주요 기능 설명 / 코드



8. 후기

1. 팀원소개

Introduction of Member



윤영광 (팀장)

- 팀원들 작업을 합쳐서 여러 체크하고 수정해서 합본 만들기
- 모든 ppt 및 보고서 작성 및 제출
- 메인페이지 Front-End 작업
 - header, footer 구분해서 만들고 간단한 애니메이션 처리, 유튜브 영상 출력
- 웹페이지 CSS 검수 및 추가사항 처리
- 스마트택배 API 이용해서 배송조회 페이지 완성
- 결제용 Import API 사용.
- 최근 본 상품 출력하는 팝업창 구현 - JavaScript 사용해서 이용자의 스크롤에 따라 같이 움직이면서 화면 밖을 벗어나지 않도록 구현.
- JavaScript 와 jQuery, ajax를 이용해 주문페이지 구현, 장바구니 페이지 및 결제 페이지 CRUD 구현 및 페이징 처리.
- 로그인 페이지 Front-End 작업 및 간단한 할수작업
- Oracle 사용해서 Database 설계 및 구축하고 MyBatis 를 사용해 DB 와 Java 연결



김동준 (팀원)

- API 사용해서 랜덤 명언 출력 작업 완료
- 레시피, 밀키트 크루링작업과 JSON 데이터 처리 JSON 파일을 Database 에 넣기
- 메인 페이지에서 최근 이벤트 팝업창 표시
 - 추가적으로 Local Storage 를 이용해 오늘은 보지 않기 체크박스 창 구현완료.
- 이벤트 페이지 구현.
 - (1) 유저들이 접속한 페이지와 정보를 실시간으로 수집해서 DB에 저장 후 가져와서 방문자 몇 명이 어느 페이지를 주로 이용했는지 나타내는 그래프 구현.
 - (2) 이용자가 문자열을 출력할 수 있는 전광판 기능 구현.
 - (3) 출석체크페이지 만들어서 포인트와 연속 출석 기능 구현.
 - (4) 일정을 기록할 캘린더 기능 구현.
 - (5) billboard.js 를 이용한 표 작성
- Github 마스터로서 작업물 관리와 Database 총괄. DB 설계 및 Oracle 사용해서 Database 설계 및 구축하고 MyBatis 를 사용해 Database 와 Java 연결 작업완료



김수진 (팀원)

- 네이버 메일 API 사용해서 회원가입 시 본인확인
 - 회원가입 시 입력부분에 정규식을 사용해 조건에 맞을 경우에만 회원가입 되도록 설정
- 지도 API 사용해서 본인 주소 확인
- 비밀번호 암호화 처리
- DB users 테이블, merchant 테이블 생성
- 회원가입, 마이페이지(개인정보 수정), 탈퇴하기 crud 구현



전병현

(팀원)

- 회원가입 기능 구현
- 자주 물어보는 질문(FAQ) 와 1:1 문의(QNA) 페이지, 그리고 공지사항(Notice) 페이지 각각 CRUD(등록, 삭제, 수정)기능 작성 완료
- 웹 통일감을 위한 고객센터 탭 안의 각 페이지에 CSS 와 JS 및 애니메이션 작업 완료.
- 고객센터 탭 안의 각 페이지마다 제목, 내용, 작성자 등의 방식으로 검색할 수 있는 keyword search 기능 구현과 각 페이지를 나타내는 페이지 처리 완료.
- 메인페이지에 ajax 를 이용해 비동기 방식으로 최근에 작성된 공지사항을 5개씩 실시간 출력하는 기능 구현
- Oracle 사용해서 Database 설계 및 구축하고 MyBatis 를 사용해 Database 와 Java 연결 작업완료



정준혁

(팀원)

- 상품 정보를 담고있는 JSON 파일 Database에 추가하기
- 웹 통일감을 위한 product 페이지에 CSS 와 간단한 애니메이션 처리 완료.
- Kakao Map API 사용해서 지도 출력하고 이용자의 현재 위치정보를 받아온 다음 근처에 있는 유명 식당들 보여주는 '내 주변 식당 기능' 구현
- 밀키트 판매 페이지(productList, productDetailList) 작성 후 파일업로드, 다운로드 기능 구현, 이후 일반 이용자들이 보는 페이지와 가맹점주(Merchant) 가 보는 페이지 구분해서 구현.
- 상품 등록, 상품 수정, 상품 삭제 기능 (CRUD) 구현 완료
- 각각의 페이지에 페이지처리 완료 후 제목, 내용, 작성자 등으로 검색 가능한 keyword search 기능 구현
- Oracle 사용해서 Database 설계 및 구축하고 MyBatis 를 사용해 Database 와 Java 연결 작업완료



조현진

(팀원)

- 레시피 JSON 파일 Database에 추가.
- 피그마 사용해서 메뉴바 만들기.
- 레시피에 알러지 및 날씨 태그 작업,
- 날씨 API 사용해서 웹페이지 상단의 원쪽 부분에 현재 날씨와 온도를 실시간으로 출력해주는 기능 구현 완료.
- 레시피 페이지에 웹 통일감을 위한 CSS 와 JS 작업 완료 후 웹페이지 전체에 이용자의 이동 경로를 나타내주는 page path 처리.
- 레시피 페이지에서 레시피 제목, 내용, 작성자 등으로 검색이 가능한 Keyword Search 기능 만들고 각각 페이지처리 완료.
- recipeList(기본 레시피 페이지)와 recipeDetailList(레시피 상세보기) 페이지에 각각 등록, 수정, 삭제 기능 구현 후 파일 업로드 및 다운로드, 파일 수정, 파일 삭제 작업 완료.
- Oracle 사용해서 Database 설계 및 구축하고 MyBatis 를 사용해 Database 와 Java 연결 작업완료

2. 선정이유

Reason for topic



나는 “점심 메뉴”를

단위 : %



점심식사 장소를 고려하는

기준

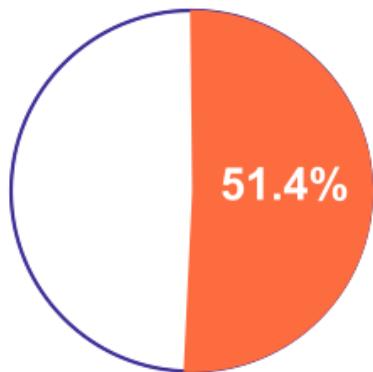
복수선택 가능



* 전국 만 19세 이상 남녀 직장인 1,700명을 대상으로 한 설문조사

결정을 내리지 못하는 순간

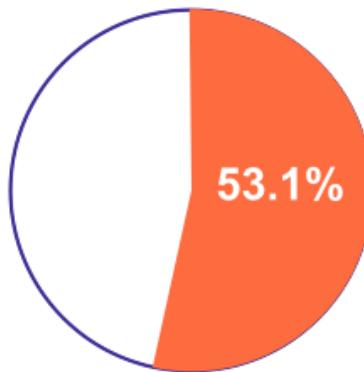
선택이 가장 어렵다



점심메뉴 선정

51.4%

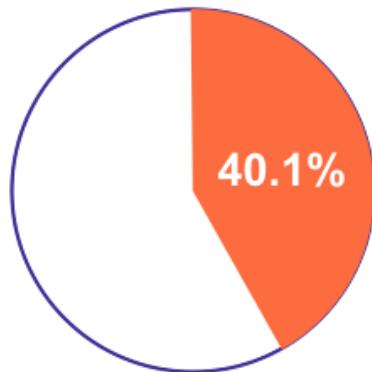
이럴 때 고민이 된다



중요한 날 옷차림

53.1%

고민 과정이 즐거울 때



점심메뉴 선정

40.1%

A middle-aged man with a white beard and mustache, wearing a white chef's toque and jacket, stands with his hand to his chin in a thoughtful pose. He is looking upwards and to the right. A large, dark gray speech bubble originates from his head, containing the Korean text "오늘 뭐먹지?".

오늘 뭐먹지?

주제선정이유

“오늘 점심 메뉴 뭐야?”, “~~씨 오늘 점심 뭐 드세요?”라는 대화는 유치원이나 초등학교에 다니는 학생들부터 대학교에 다니고 직장에 다니고 있는 성인들까지 식사 시간이 다가오면 모두가 하는 이야기입니다.

이처럼 무엇인가를 먹고 영양분을 보충한다는 의미를 가진 ‘식(食)’이라는 단어는 사람의 인생에서 굉장히 중요한 요소입니다.

또한 ‘**도락(食道樂)**’이라는 표현이 있는 것 처럼 요즘에는 무엇인가를 먹는다는 행위가 단순히 영양분을 보충하는 행위가 아니라 누군가에게 행복과 감동을 전하기도 합니다.

‘퇴근하고 집에 가서 뭐 먹지?’, ‘오늘은 □□가 땡기는데 뭐랑 같이 먹으면 좋을까?’라는 생각을 해본 적 있으신가요?

이런 생각은 성별, 나이 관계없이 누구나 한 번쯤 생각하는 내용입니다.

우리는 메뉴선정에 대한 고민 때문에 하루에 점심, 저녁 최소 10분 정도씩 불필요한 에너지를 소모합니다.

따라서 사람들은 이런 고민을 줄이기 위해 다양한 어플리케이션과 웹페이지의 도움을 받습니다. 그러나 직접 사용해본 결과 각각의

어플리케이션과 웹페이지에는 사용하는데 있어서 큰 불편함을 제공하는 뚜렷한 단점들이 꽤 많이 존재했습니다.

그래서 우리가 느낀 단점을 최대한 보완하여 현대인들의 메뉴고민 시간을 최대한 짧고, 최대한 행복하게 만들어드리는 것이 목표입니다.

‘오늘 하루 수고했는데 맛있는 음식 먹고 힘내야지!’

와 같은 생각이 들었을 때 **cookology** 를 사용해 주신다면, 신뢰도 높은 정보들과 맛있는 음식과 으로 보답하겠습니다.

유사사이트 분석



서비스 소개 모바일 APP 서비스

현 위치 기반 근거리 내 식당
현재 내 위치 주변 최대 15분 거리 내에
있는 식당 리스트 확인



미리 주문하자!

걸어서 최대 15분 거리 내 맛집
빠르게 미리 주문하고 기자!



이제 식당에 가서 주문하고 기다리시지 마시고,
식당에 가기 전에 미리 주문과 결제, 자리 확보까지
한 번에 하세요.

[더 보기](#)

MINJUN KIM
개발자

평가 및 리뷰

5.0



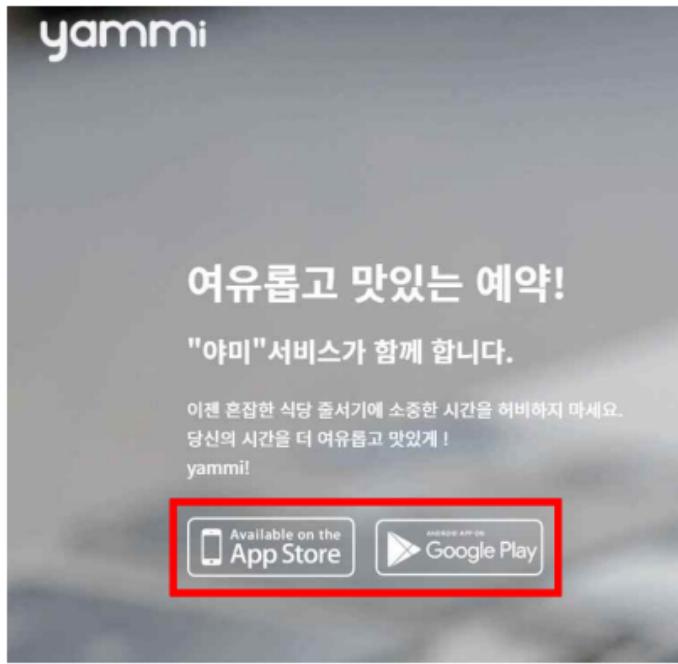
2개의 평가

새로운 기능



버전 기록

유사사이트 분석



A Google search results page for '야미 음식주전'. The first result is a link to the website <https://www.siksinhot.com>, which is described as '야미아미와주먹밥|맛집, 한식맛집, 식신 대한민국 No.1 맛집검색 ...' (Yamiamei and Siksikhap | Restaurant, Korean restaurant, No.1 restaurant search in South Korea ...). The second result is another link to the same website, also describing it as a top restaurant in South Korea.

<https://www.siksinhot.com> > ... ▾
야미아미와주먹밥|맛집, 한식맛집, 식신 대한민국 No.1 맛집검색 ...
맛집 야미아미와주먹밥은(는) 베스트 맛집으로 회식, 모임, 데이트, 추천맛집 입니다. ... 한국음식
> 떡볶이라면/분식, 전화번호. 051-747-2992. 주소.

<https://www.siksinhot.com> > ... ▾
야미|맛집, 양식맛집, 식신 대한민국 No.1 맛집검색, 맛집추천
맛집 야미는 Premium 수제 생 돈까스, 수우노우지스 돈까스, 야미정식!, 알밥, 크랩알밥, 눈꽃치즈 알
밥, Take out, 훈제연어덮밥, 새우튀김우동, 야미정식!, 스노우치즈크랩알밥, ...

<http://m.blog.naver.com/syjmhot> ▾
야미가 메뉴 추천 ; 연어회덮밥 & 돈까스 - 네이버 블로그
2017. 7. 11. — 야미가에서 시켜먹은 연어회덮밥 & 돈까스 원래는 침치회덮밥 먹고 싶었는데 없
어서 ; 야미가 메뉴 추천합니당!! 연어회덮밥이 메인인구요~ ; 단무지랑 김치.

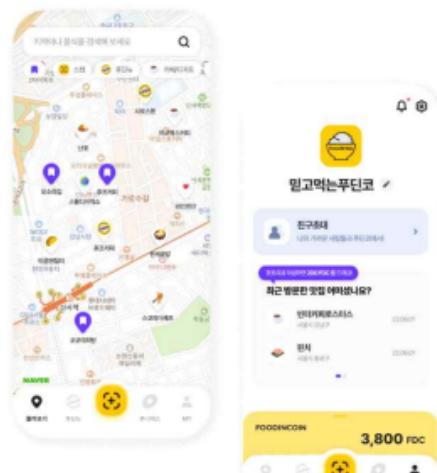
<https://blog.naver.com/yahohoo00> ▾
배달음식 중 최고 야미가 실패 없는 메뉴선택은 - 블로그
2016. 6. 2. — 배달음식 편하고 좋는데 실패 없이 먹으려면 야미기가 답이다 ㅋㅋ 비주얼 딱 사진
그대로 오고~ ... 스테이크덮밥 추천 메뉴어서 먹어봤는데~

유사사이트 분석

foodinko

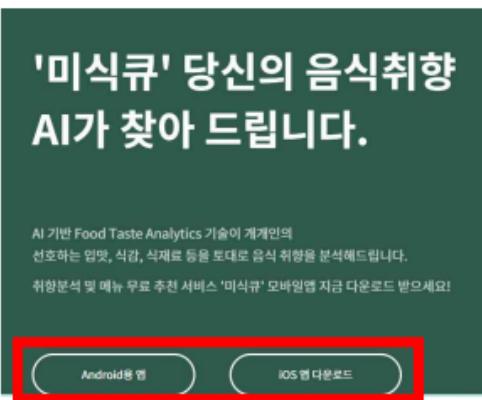
밀고 먹는 푸드코

80만 SNS 구독자가 인정한 대한민국 맛집의 기준!
이제 지도 앱에서 편하게 보며 다니세요.



평가 및 리뷰	
앱이 자꾸 범위요 ★☆☆☆☆	11월 9일 Min_e
조금 이동하면 업주고 앱 중요하고 다시 조금 이동하 면 엄청고... 무한 반복이에요	
개발자 답변 인녕하세요, Min_e님! 이용에 불편을 드려 죄송합니 다. 번거로우시겠지만 사용하시는 기기정보 더 보기	11월 16일
로그인도 계정찾기도 다안돼요 ★☆☆☆☆	8월 16일 p9206845
모듈인가요 계속 아무것도 안되는데 서비스 일시적인 문제라고;; 시작을 못하겠어요 앱	
개발자 답변 안녕하세요,p9206845님! 푸드코 앱 이용에 불편을 드려 죄송합니다. 일시적인 오류가 있었던걸 더 보기	8월 31일
앱 업데이트 하니깐 안켜져요 ★☆☆☆☆	8월 3일 나나난나나느느
커지게해주시여ㅠㅠ	
개발자 답변 나나난나나느느느느님, 안녕하세요! 이용에 불편을 드려 죄송합니다.	8월 4일 더 보기
왜 실행이 안되죠?	8월 2일

유사사이트 분석



Google

고잇

전체 이미지 지도 동영상 뉴스 더보기 도구

검색결과 약 18,600개 (0.39초)

[주식회사 고잇\(고잇\) 기업정보 - 넥스트유니콘](https://www.nextunion.com.kr/company)
고잇, 직장인 맞춤 메뉴 추천, #직장인, #점심, #메뉴추천, #고잇, #취향, #음식, #밥, #점심메뉴, #맞춤, #회식메뉴, #메뉴고민, #워터머지?, #외식, #맛집추천, #예약,

[고잇 - 실시간 식당 예약 서비스 4+ - App Store](https://apps.apple.com/app/고잇-실시간-식당-예약...)
2022. 2. 26. ... 고잇 - 실시간 식당 예약 서비스 4+ - 실시간 식당 예약/주문 맞춤 메뉴 추천 서비스
... - 고잇 iPhone - 고깃국 - 블로그 - 새로운 기능 - 평가 및 리뷰 - 경비 ...
★★★★★ 평점: 3.9 · 리뷰 7개 · 무료 · iOS

Google

거꾸로 레시피

전체 이미지 동영상 뉴스 지도 더보기 도구

검색결과 약 774,000개 (0.35초)

[App Store에서 제공하는 거꾸로레시피 - Apple](https://apps.apple.com/app/거꾸로레시피)
2021. 8. 18. ... 거꾸로레시피 4+. 보유한 재료로 바로 요리할 수 있는 레시피. Chaeun Lee.
iPhone용으로 디자인됨.
★★★★★ 평점: 3.7 · 리뷰 99개 · 무료 · iOS

Google

우리뭐먹지

전체 이미지 동영상 뉴스 지도 더보기 도구

검색결과 약 182,000,000개 (0.42초)

[App Store에서 제공하는 우리뭐먹지 - Apple](https://apps.apple.com/app/우리뭐먹지)
2022. 7. 21. ... 다 같이 뭐 먹을지 고민될 때.. 우리뭐먹지.. 그날 회식메뉴 찾다가 친한 직장인이 만...
...을 때마다 뭘 먹어야 할지 고민되는 그날 회식메뉴 찾다가 친한 직장인이 만...
★★★★★ 평점: 5 · 리뷰 8개 · 무료 · iOS

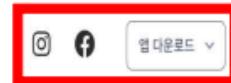
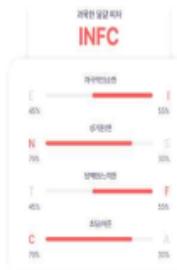
유사사이트 분석

fooiy

내 입맛은
무엇일까?
푸이티아이

내 푸이티아이는 무엇일까요?

입맛에 맞는 음식점을 발견해 보세요!



푸이

검색결과 약 860,000개 (0.37초)

<https://ko.wikipedia.org/wiki/푸이>

푸이 - 위키백과, 우리 모두의 백과사전

푸이 ; Emblem of the Emperor of Manchukuo.svg. 만주국 황제 ; Emblem of the Emperor of Manchukuo.svg. 만주국 황제 - 1934년 3월 1일 ~ 1945년 8월 18일 본인(대동왕).



사망지 : 중화인민공화국 베이징

가문: 아이신 고로 할아버지

연호: 청 제국: 선통(宣統) 만주어: 푸이 ...

기타 친인척: 순현친왕 혁현(조부), 순현...

<https://namu.wiki/s선통제>

선통제 - 나무위키

4일 전 — 조선의 이방원, 이성계와 비슷하게 정식 칭호보다 푸이(溥儀)라는 본명으로 잘 알려져 있다. 그리고 마지막 황제라 하여 일제(末帝)라고도 일컬어진다.

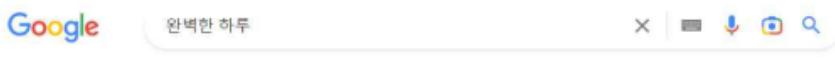
마지막 황제 · 효각민황후 · 장춘복벽 · 순친왕

유사사이트 분석



<https://woman.donga.com/BestClick/all>
냉장고 파역기, 그 고귀함에 대하여 - 여성동아
 2022. 8. 8. — 냉장고 파역기를 시작하게 된 데 '작은 냉장고의 공도 빼놓을 수 없다. 신혼집 옵션으로 떨어진 냉장고는 냉장, 냉동 두 칸으로 이루어진 작은 크기였다.'

<https://blog.naver.com/cardfkorea>
냉장고 파역기도 방법이 따로 있다? 지금 당장 시작하는 ... - 블로그
 2021. 7. 2. — 둘다리도 두들겨 보는 심정으로 냉장고 파역기 끌립들을 하나하나 속지해 보세요, 오리 내공은 높아지고, 지갑이 두둑해지고, 냉장고가 가벼워지며, 기분 ...



<http://www.yes24.com/Product/Goods>
완벽한 하루 - YES24
 2016. 2. 1. — 성취의 기쁨으로 전을하는 '완벽한 하루'가 펼쳐진다! 오늘 하루, 무엇을 위해 살았는가? 아무리 열심히 바쁘게 살아도, 이 질문에 속 시원히 대답할 수 ...
 ★★★★★ 평가: 9.2/10 · 리뷰 55개 · ₩13,500



<https://m.blog.naver.com/cutesiwoo>
오늘 저녁 뭐 먹지? 저녁메뉴 추천 레시피 22가지 - 네이버 블로그
 2017. 11. 5. — 1. 두부조림, 저녁 준비될 여유가 없을 때 후디식 만들어낼 수 있는 법도둑 빈집이에요. · 2. 어묵탕(모방탕). 후위진 날씨에 딱 어울리는 국물요리 · 3.

<http://완벽한하루.com>
완벽한 하루 - 완벽한 하루를 보내는 방법 :: 데이트 · 관광 코스 ...
 당신의 완벽한 하루를 만들어 드립니다. 편리한 데이트 코스 추천 앱. 편리한 데이트 코스 추천 앱. 맞춤 코스 제작. 원하는 장소에서 원하는 조건으로,

<https://speedwagwon.tistory.com>
오늘 저녁 뭐 먹지 | 저녁 메뉴 추천 | 저녁식사 메뉴 고르기 모음
 2021. 6. 16. — More videos on YouTube · 미국국, 해장국, 선지혜장국, 소고기붓국, 순댓국, 콩나물국, 뚝배기불고기, 찬복命运기, 고리김밥, 닭곰탕, 삼계탕, 맥주, ...

유사사이트 분석 : 문제점

- 조사자료 대부분 어플리케이션 사용 최적화에 중점을 두고 있다.

웹페이지 존재하지 않음	5개
웹페이지는 존재하지만 별다른 기능없음 - 어플리케이션 다운로드 페이지 링크만 존재	6개
웹페이지 존재 + 현재도 기능(식단, 음식추천) 사용가능함 - 단점1 : 데이터 적음, 가시성 없음 - 단점2 : 폐업식당 정보기재 (지속적인 정보관리 없음) - 단점3 : 무분별한 광고 재개	4개

유사사이트 분석 : 문제점

1. 사진 배열이 정신없어 보인다.
2. 음식의 맛이 없어 보인다.



유사사이트 분석 :

알러지알려줘 - 식품 알레르기 대처 유리
아이들 위한
알러지알려줘



설명보기 →

<https://koreascience.kr/article/CFKO202121751587342.pdf>

<https://play.google.com/store/apps/details?id=me.allergy.i>

장점	알러지 성분 입력 시 섭취가능 식품 파악가능
단점	앱만 제공 (접근성 저하)



Artificial Intelligence and Deep Learning Solutions for Food Recognition

We automate, optimize and provide an added value to processes related to food and nutrition.



<https://logmeal.es/>

장점	음식 추천시스템(API 제공함) - 이미지 인식해 음식판별 후 기록함
단점	웹만으로는 구현불가 = 앱만 사용가능

유사사이트 분석 :

문제점

The screenshot shows the homepage of Menupan, a Korean cooking recipe website. At the top, there's a navigation bar with links like '인기', '최신', '인기', '인기', and '전체카테고리'. Below the navigation is a large search bar. The main content area features several recipe cards with images and titles. One card for '드라이브' (Drive) has a green banner at the bottom. Another card for '미국적인 매력이 느껴지는 아메리칸 다이닝 타코, 뉴치즈 맛집 리스트!' (List of American-style tacos and nachos) includes a small video thumbnail. The bottom section shows a grid of smaller recipe cards, one of which is for '김치 냄비' (Kimchi Stew). There are also sections for 'MENUPAN TIMES' and 'Menupan 뉴스'.

<https://www.menupan.com/Cook/>

장점	레시피 각 항목 맷글, 대댓글 작성 가능
단점	사이트 인터페이스 구식

The screenshot shows a search results page for '요리백과' (Cooking Encyclopedia) on Naver. The top navigation bar includes '검색어', '날씨', '날씨', '날씨', and '날씨'. Below the search bar, there's a link to '제작자' and a button to '로그인'. The main content area shows a list of search results with titles like '요리백과' (Cooking Encyclopedia), '요리백과' (Cooking Encyclopedia), and '요리백과' (Cooking Encyclopedia). Each result has a thumbnail image, the title, and a snippet of text. The bottom of the page shows a sidebar with links for '주한조선족', '최신', '기사나온', and a '마이페이지' link.

<https://terms.naver.com/list.naver?cid=48156&categoryId=48156>

장점	없음
단점	접근성 저하 사용자가 원하는 정보도출 불가, 불편

유사사이트 분석 :

문제점

< 검색

고객
개발자

평가 및 리뷰

3.9

(최고 5점)



7개의 평가

새로운 기능

비전 2.0.3

비전 기록

1년 전

* 지역 선택 기능 추가

앱이 수집하는 개인정보

고객 개발자가 아래 설명된 데이터 처리 방식이 앱의 개인정보 보 처리방침에 포함되어 있을 수 있다고 표시했습니다. 자체 한 내용은 [개발자의 개인정보 처리방침](#)을 참조하십시오.



사용자에게 연결된 데이터

다음 데이터가 수집되어 신원에 연결될 수 있습니다.

❶ 연락처 정보

앱이 자꾸 멈춰요

★☆☆☆☆

11월 9일

Min_e

조금 이동하면 멈추고 앱 종료하고 다시 조금 이동하면 멈추고.. 무한 반복이에요

개발자 답변

11월 16일

안녕하세요, Min_e님! 이용에 불편을 드려 죄송합니다.
다. 번거로우시겠지만 사용하시는 기기정보 [더 보기](#)

앱으로 검색한 밤집으로 찾아갔는데, 영업을 중단한지 조금 되어보았습니다. 또, 그저께는 비슷한 위치에서 다른 밤집을 찾아갔는데, 위치가 옮바르지 않는건지 밤집이 보이지가 않았습니다. 이런 일이 자주 일어나면 다른 친구들을 데리고 자신있게 '여기 가자' 하며 이끌거나 추천하기가 힘듭니다. 저 혼자 활용할 때도 이 집 지금 있긴 하나 라는 생각이 먼저 들 정도로 신뢰가 떨어집니다. 신뢰감을 높이기 위해 데이터를 비롯해 조속한 업데이트가 필요하다고 생각됩니다.

개발자 답변

4년 전

케미주원님, 안녕하세요. 빅데이터 맛집검색, 다이닝 코드입니다. [더 보기](#)

저번에도 적었는데 제발 봐주세요

★★★★☆

7월 26일

围观组 이해가

맛집리뷰하는 어플 중 가장 선호합니다

거리순 최신순 등으로 지도에 표시가 되어있는데

추가로 내가 평가한 맛집도 지도에 전체적으 [더 보기](#)

개발자 답변

8월 3일

围观组 이해가님 안녕하세요, 다이닝코드입니다. 우선 답변이 늦어진 점 대단히 죄송합니다. [더 보기](#)

로그인도 계정찾기도 다안돼요

★☆☆☆☆

8월 16일

p9206845

오류인가요 계속 아무것도 안되는데 서비스 일시적인 문제라하고;; 시작을 못하겠어요 앱

개발자 답변

8월 31일

안녕하세요,p9206845님! 푸드코 앱 이용에 불편을 드려 죄송합니다. 일시적인 오류가 있었던걸 [더 보기](#)

앱 업데이트 하니깐 안켜져요

★☆☆☆☆

8월 3일

나나난뉴는느느

커지개해즈시여ㅠㅠ

개발자 답변

8월 4일

나나난뉴는느느 님, 안녕하세요! 이용에 불편을 드려 죄송합니다. [더 보기](#)

유사사이트 분석 :

MangoPlate >
개별자

평가 및 리뷰 모두 보기

3.3 (최고 5점)



327개의 평가

일반하세요? 2년 전 ★☆☆☆☆ 리리리리리리랄

없어진 식당들 왜 그대로 두나요? 이게 대체 몇번째
에어? 없어진지 있는지 믿고 쓴 어플인데 와이러시
는 거예요? — 매번 허탕치고 돌아서고 진 더 보기

개발자 답변 2년 전 안녕하세요. 망고플레이트입니다. 이용에 불편을 드려
안타깝습니다. 전달주신 피드백은 개선하여 더 보기

새로운 기능 버전 기록

버전 1.6.58 11개월 전

MangoPlate는 더 좋은 서비스를 제공하기 위해 꾸준히 업데이트를 진행하고 있습니다. 업데이트 때마다 안정성을 개선하고 있습니다.

신뢰가 가지 않아 탈퇴했습니다. 10월 13일

★☆☆☆☆ 초록수집가

몇년간 잘 사용했으나 사용할수록 형편성이 맞지 않은
리뷰와 유추하기 어려운 그들만의 알고리즘, 입맛대로
처리하는 cs운영방식 등 리뷰 사이트의 갑질로 느껴
짐과 동시에 더이상 신뢰가 가지 않아 탈퇴합니다. 번
창하세요.

고객센터가 있긴한가요? 8월 23일

★☆☆☆☆ 공통느낌

태어나서 이런 리뷰 창피해서라도 안쓰는 편입니다
가게 측 잘못으로 구매한 잊딜이 사용하지 않은게 사
용했다고 처리가 되었고 음식값을 따로 온전히 지불했
습니다

가게측에서도 망고플레이트 고객센터에 문의해보라
고 말안하고 가게를 나갔습니다

망고플레이트는 고객센터 자체가 운영하지 더 보기

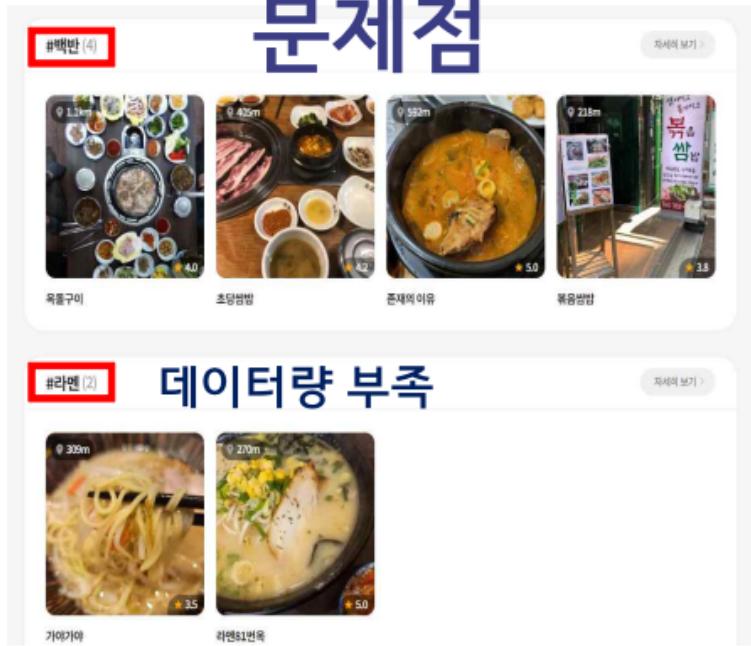
왜 엉터리 식당만 선정하세요? 10월 2일

★☆☆☆☆ 망고플레이트국힘

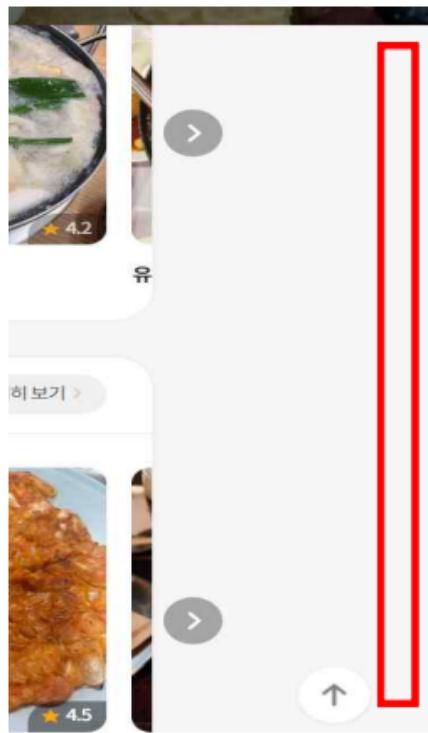
망고플레이트를 통해 좋았던 맛집 찾아본 적이 없었습니다. 평점 4.8 이라는 맛집이 음식 조리 상태도 않좋았고 직원 서비스 수준도 않좋고 찾기 힘든 곳에 위치해 있고...

이런 식당들이 한두곳도 아니고 수도권, 지방 가리지 않았습니다. 변화가, 주거지역 상관없이 다 더 보기

유사사이트 분석 : 문제점



스크롤 없음



단점 : 불필요한 광고, 내용

만개의레시피

추천 분류 평점 메가진

제 157회 만개의레시피 공연한 두부요리 레시피 알려주세요.

쇼핑 BEST

- 플라보정 세라믹크롬 프라이팬
- 음식통이 격파되는 싱크대 배수
- 아이닉 차이슨 무선청소기 new
- 포기 4구 6구 이중필

19,800원 17,900원 179,000원 16,900원

해골린 만개의 BEST 요리모음

디시나 디시 스포츠경기

Google Play 게임즈 베타 체험

Google Play 게임즈 베타를 통해 모바일 게임을 탐색하고 플레이하세요

다운로드하기

블로그리뷰

구리 2022년 9월 18일

장단점 분석 : 다이닝코드

장점	없음
단점	<p>추천이 아닌 사용자가 직접 검색해서 정보파악 해야함 회원가입 방법이 휴대전화, 페이스북만 존재함 광고가 너무 많다</p>



신규 회원가입

다이닝코드에서는 회원님의 보안 강화 및 편리한 서비스를 제공해

드리기 위해 휴대전화 인증을 하고 있습니다.

휴대전화 인증 가입

맛집검색, 좋아요, 비밀번호찾기, 정보받아보기, 평가작성, 포인트 적립 및 사용, 다코박스 상품구매, 할인쿠폰 등의 모든 서비스 혜택을 받을 수 있습니다.

페이스북 가입

페이스북 신규 가입자도 휴대전화인증이 필요하며, 가입 후 비밀번호 없이 로그인하실 수 있습니다.

휴대전화 회원가입

페이스북 회원가입

[로그인 하기](#)

장단점 분석 : 망고플레이트

장점	커다란 사진 이용자들의 솔직한 리뷰
단점	음식명, 지역 등을 구체적으로 입력해야함

ANGO LATE

지역, 식당 또는 음식

추천 검색어

- 강남역
- 용리단길
- 유류버주천
- 신촌

인기 검색어

최근 검색어

2022-08-09
애매하게 마감 시간에 방문해서 웨이팅 없이 바로 들어가서 클램차우더랑 쉬림프 파스타 미트볼 주문했어요.

음 너무 기대를 많이 했는지 쉬림프 파스타는 맛있게 먹었는데 클램차우더는 저한테는 조금 무거운 느낌이였고 미트볼 파스타는 파스타면에서 너무 아무맛이 안나서…저는 소스와 면이 어우러지는 파스타를 좋아하나봐요.

기대보다는 조금 아쉬웠지만 매장이 바쁜데도 직원분들 너무 친절하시고 매점 분위기도 좋아서 좋은 시간이였어요



포털 소개지



더 알아보기



A middle-aged man with a white beard and mustache, wearing a white chef's toque and jacket, is shown from the chest up. He is looking upwards and to his right with a thoughtful expression, his right hand resting against his chin. A large, dark gray speech bubble originates from his head, containing the Korean text "오늘 뭐먹지?".

오늘 뭐먹지?

Cookology !



3. 사용기술

Feature and Tools Used



프로젝트 사용기술

1. 팀원 소개 2. 프로젝트 기획 3. 사용기술 4. 기능정의서

FRONT-END

billboard.js



HTML



JS



CSS



AJAX
(Asynchronous JavaScript and XML)



Calendar
TypeScript



TOOLS



Bootstrap



eclipse

Visual Studio Code



python



git

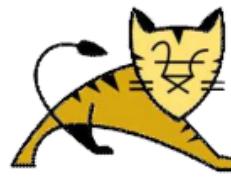
GitHub



Maven™



BACK-END



Apache
Tomcat



MyBatis

spring
MVC



ORACLE
D A T A B A S E



4. 기능정의서

Functional definition



프로젝트 사용기술

1. 팀 원 소개 2. 프로젝트 기획 3. 사용기술 **4. 기능정의**

액터	요구사항	요구사항 내역
일반회원	로그인	일반회원의 웹 페이지의 기능을 이용하기 위해 로그인 해야 한다.
	마이페이지	일반회원의 개인정보, 자신에 대한 어려거지 정보확인 가능한 마이페이지를 확인할 수 있다.
	장바구니	담기, 삭제 등을 이용하여 원하는 상품을 장바구니에 담거나 삭제 할 수 있다.
	문의	일반회원은 관리자, 가맹점에 대해서 문의를 신청할 수 있다.
	게시글	일반회원 게시글 페이지에서 Q&A, F&Q를 조회, 작성, 수정, 삭제 할 수 있다.
	리뷰	구매한 상품에 대해서 리뷰를 작성, 작성할 수 있다.
	주문결제	장바구니에 담아 놓은 상품 및, 하나의 상품에 대하여 결제 할 수 있다.
	이벤트	음식 선택에 관한 오락 프로그램을 이용할 수 있다 (ex : 랜덤플랫)
	검색	상품, 레시피를 검색하고 선택할 수 있다.
	요리선택	직접요리, 상품 배달 받을 것인지에 대한 선택지를 선택 할 수 있다.
추천		미세먼지, 성별, 날씨 관련해서 음식에 대한 추천을 받을 수 있다.
		주변 지도를 이용하여 식당을 추천 받을 수 있다.
배달		주문한 상품을 배달 받을 수 있다. 실시간 배달 현황을 조회할 수 있다.

액터	요구사항	요구사항 내역
관리자	로그인	관리자의 웹 페이지의 기능을 이용하기 위해 로그인 해야 한다.
	컨텐츠 관리	홈페이지에 대한 모든 컨텐츠를 확인할 수 있다.
	게시글	관리자는 공지사항 게시글 페이지에서 공지사항을 작성, 수정, 삭제 할 수 있다.
	문의	일반회원, 접주에 관련된 문의 신청을 허가하고, 1대1로 답변을 할 수 있다.
	회원관리	홈페이지에 가입된 일반회원, 접주에 대하여 회원 목록을 조회하고, 정지, 삭제, 관리 할 수 있다.
	카테고리 관리	홈페이지 각종 카테고리에 관련하여 추가, 수정, 삭제 할 수 있어야 한다.

액터	요구사항	요구사항 내역
비회원	회원가입	액티 각각의 기능을 사용하기 위해, 일반회원, 접주로 회원가입 해야 한다.
	장바구니	담기, 삭제 등을 이용하여 원하는 상품을 장바구니에 담거나 삭제 할 수 있다.
	주문결제	장바구니에 담아 놓은 상품 및, 하나의 상품에 대하여 결제 할 수 있다.
	검색	상품, 레시피를 검색하고 선택할 수 있다.
	요리선택	직접요리, 상품 배달 받을 것인지에 대한 선택지를 선택 할 수 있다.
	추천	미세먼지, 성별, 날씨 관련해서 음식에 대한 추천을 받을 수 있다.
	배달	주문한 상품을 배달 받을 수 있다. 실시간 배달 현황을 조회할 수 있다.

액터	요구사항	요구사항 내역
가맹점주	마이페이지	관리자의 웹 페이지의 기능을 이용하기 위해 로그인 해야 한다.
	게시글	접주 문의 게시글 페이지에서 문의 글에 대한 조회, 작성, 수정, 삭제 할 수 있다.
	로그인	접주회원의 웹 페이지의 기능을 이용하기 위해 로그인 해야 한다.
	판매	식재료 취급 접주들의 상품을 올리는 페이지 + 모리(메뉴) 소개 하는 페이지에서 일반회원들이 상품을 선택 후 구매 할 수 있다.
	상품관리	접주가 상품 판매 페이지에 등록해 놓은 상품을 수정, 삭제, 추가 할 수 있게 관리할 수 있다.
	배송	비회원, 일반회원이 구매한 상품에 대하여 배송 할 수 있다.
	매출관리	Day, week, Month에 대한 매출을 관리할 수 있다.

개발일정

Development schedule



02.28

프로젝트 기획 및 보고서 제출

팀 이름, 프로젝트 주제 및 이름, 세부 구현 기능 정리, 유사 사이트 분석,
작업 분담, 개발 툴 및 버전 관리 툴 정리, Usecase Diagram 다이어그램 작성

03.02

프로젝트 설계 회의

회의 및 설계결과물 작업 진행

03.06

UI 설계 및 보고서

각 쟁별 디자인 및 배치, 컴포넌트 연결 기능 설명
Wireframe 작성, 각자 본인 기능의 화면설계 작성

03.13

DB 설계 및 보고서

제이터를 저장 처리하기 위한 테이블, 제1정규화, 제2정규화 작성,
각자 본인 기능의 DB설계 작성, 팀장이 테이블 관계 정리해서 보고서로 제출

개발일정

총 52일 간의 개발 일정

5. 개발일정

6. Usecase Diagram 7. 주요기능 / 코드 8. 마무리

03.20

클래스 설계 및 보고서

제출, Package, Sequence Diagram 작성

Controller Method Mapping Table 작성 (사용할 오픈 api 조사해서 해당 내용 포함)
본인 기능의 Class 설계 작성 후 팀장이 Mapping url 확인해서 보고서로 제출

03.27

프로젝트 구현 및

회기 4/1~ 4/12(수)

1일 1기능 페이지 완성을 목표로 매 일마다 팀장이 일일 작업일지 작성후 제출 하기.
1일 2회(오전, 오후) 프로젝트 통합 관리 : Merge 시 backup 파일 반드시 만들기!

04.13

최종 테스트

전체 기능 Beta Test 진행 후 디버깅.

04.14

프로젝트 발표

프로젝트 발표 보고서 ppt/pdf 로 작성해서 제출하기

프로젝트 최종 결과물 zip 압축 제출

개발일정

총 52일 간의 개발 일정

6. Usecase Diagram

Usecase Diagram

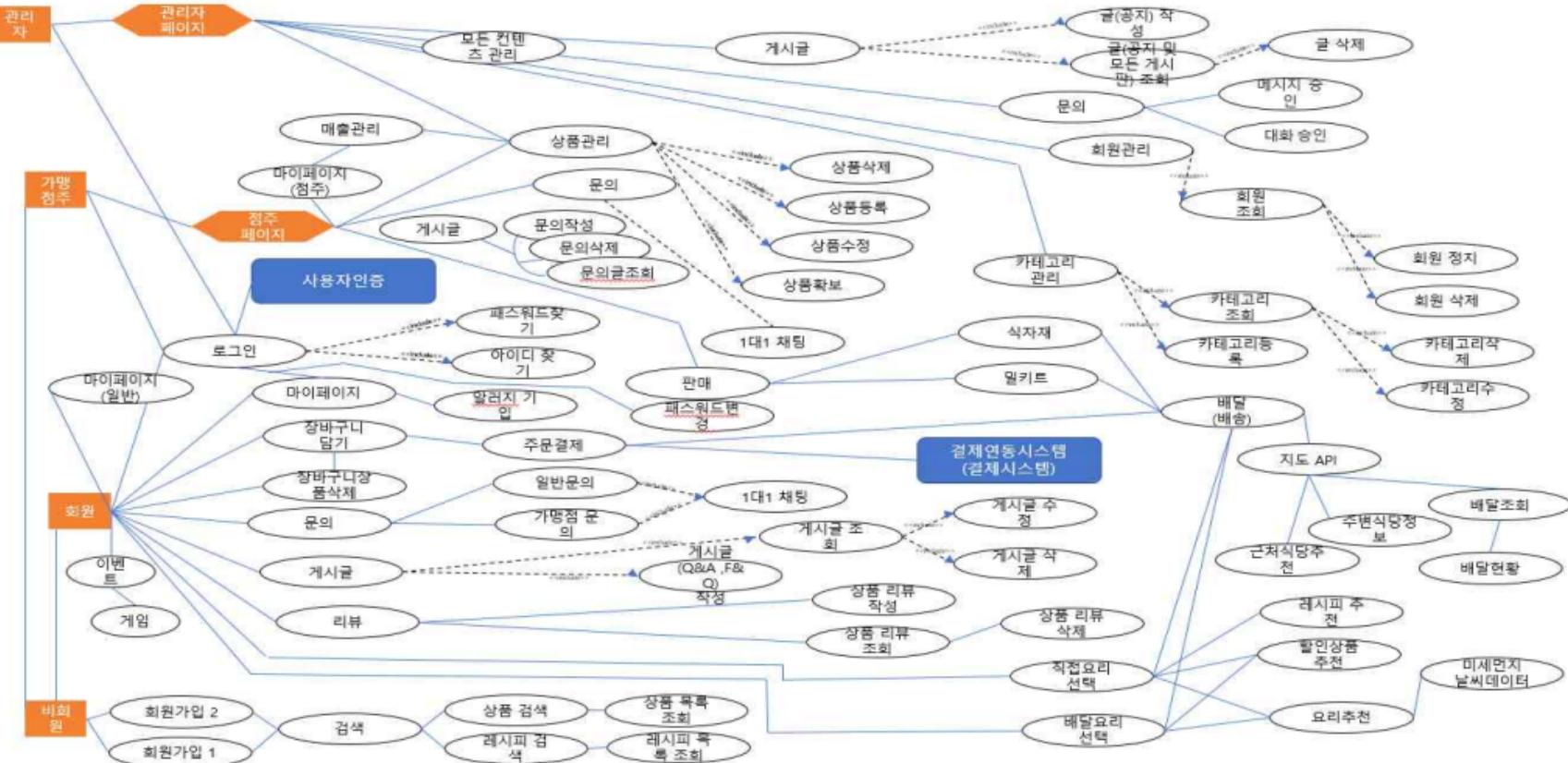


유스케이스 다이어그램

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드 8. 마무리



7. 주요 기능 / 코드

7-1. 김수진 : 회원가입 , 마이페이지

Main Features and Codes

1. 회원가입

비밀번호 유효성 검사 + 중복검사

```
//비밀번호 유효성 검사
function validatePwd() {
    let pwdInput = document.getElementById("users_password");
    let users_password = pwdInput.value;
    let message = document.getElementById("pwdValidateError");

    let pwdRegex = /^(?=.*[a-zA-Z]).*[a-zA-Z].*[a-zA-Z].*[a-zA-Z](?=.*d.*d)(?=.*[!@#$%^&_]).{7,}$/;

    if(pwdRegex.test(users_password)){
        message.innerHTML = "사용 가능한 비밀번호입니다.";
        flag = true;
        console.log(flag);
        message.style.color = "green";
    }else{
        message.innerHTML = "사용 가능한 비밀번호입니다.";
        message.style.color = "red";
        flag = false;
        console.log(flag);
        return false;
    }
}

//비밀번호 중복검사
function checkPassword(){
    let users_password = document.getElementById("users_password").value;
    let users_password2 = document.getElementById("users_password2").value;

    if(users_password == users_password2){
        document.getElementById("passwordError").innerHTML = "비밀번호가 일치하지 않습니다.";
        passwordError.style.color = "red";
        flag = true;
        console.log(flag);
        return false;
    }else{
        document.getElementById("passwordError").innerHTML = "일치하는 비밀번호를 확인했습니다.";
        flag = false;
        console.log(flag);
        passwordError.style.color = "green";
    }
}
```

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

Cookology

Mail(ID)

비밀번호 구성조건

- 영문자 대,소문자 관계없이 4개 이상 포함해주세요
- 숫자는 2개 이상, 특수문자 1개 이상 포함해주세요
- 비밀번호는 7 ~ 20자의 길이로 구성해주세요

비밀번호

사용 가능한 비밀번호입니다.

비밀번호 재확인

입력하신 비밀번호를 확인했습니다.

네이버메일 인증 API

```
var code; //이메일전송 인증번호 저장위한 코드

$(function(){
    /* 등록번호 이메일 전송 */
    $(".mail_check_button").click(function(){
        alert("인증번호 발송되었습니다. 인증번호를 확인해주세요.");
        var email = $(".mail_input").val(); // 등록된 이메일
        var checkBox = $(".mail_check_input"); // 등록번호 입력란
        var boxWrap = $(".mail_check_input_box"); // 등록번호 입력란 부스
        $.ajax({
            type:"GET",
            url:"mailCheck.do?email=" + email,
            success:function(data){
                checkBox.attr("disabled",false);
                boxWrap.attr("id", "mail_check_input_box_true");
                code = data;
            }
        });
    });
}); //document.ready

//이메일인증번호 확인
function emcheck(){
    var inputCode = $(".mail_check_input").val(); // 입력코드

    // 허용되는 경우
    if(code != null && (inputCode == code)){
        alert("인증번호가 일치합니다.");
        return true;
        // 일치하지 않을 경우
    } else {
        alert("인증번호를 다시 확인해주세요.");
        return false;
    }
}
```

1. 회원가입

전화번호 유효성검사

```
function validatePhone(){
    let phoneInput = document.getElementById("users_phone");
    let users_phone = phoneInput.value;
    let message = document.getElementById("phoneValidationMessage");

    let phoneRegex = /^\d{11}$/;

    if(phoneRegex.test(users_phone)){
        message.innerHTML = "전화번호를 확인했습니다.";
        flag = true;
        console.log(flag);
        message.style.color = "green";
    }else{
        message.innerHTML = "전화번호 형식에 적합하지 않습니다";
        message.style.color = "red";
    }
    flag = false;
    console.log(flag);
    return false;
}
```

우편번호 API

```
// 우편번호 API
$(function(){
    $('#postcodify_search_button').postcodifyPopUp();
});

$(function execute(){...});
```

이름

김수진

이름을 확인했습니다.

전화번호

01056455127

전화번호를 확인했습니다.

우편번호

12419

검색

주소

경기도 가평군 가평읍 가평천길 4-17

(대곡리)

알려지 정보

- 난류 우유, 유제품 냉동 견과류 밀 칡깨
- 콩(대두) 과밀 및 제초 해산물 및 조개류 육류

회원가입 완료

홈페이지로 돌아가기

이름 유효성검사

```
function validateName(){
    let nameInput = document.getElementById("users_name");
    let users_name = nameInput.value;
    let message = document.getElementById("nameValidationMessage");

    let nameRegex = /^[ㄱ-ㅎㅏ-ㅣㅏ-ㅣ]+$/;

    if(nameRegex.test(users_name)){
        message.innerHTML = "이름을 확인했습니다.";
        flag = true;
        console.log(flag);
        message.style.color = "green";
    }else{
        message.innerHTML = "한글, 영어(대, 소문자)를 조합해 3-10글자까지의 작성 가능합니다.";
        message.style.color = "red";
        flag = false;
        console.log(flag);
        return false;
    }
}
```

최종검사

```
$(function () {
    console.log($('#usersSignup').serialize());
    $('#testSubmit').on('click', () => {
        const form = $('#form').serializeObject();
        if(flag == true){
            $.ajax({
                url: "usersSignup.do",
                type: "post",
                data: form,
                success(result) => {
                    console.log(result.status);
                    console.log("success");
                    window.location.replace("main.da");
                },
                error(function(result, status, errorData){
                    console.log("error code : " + result.status
                        + " errorMessage : " + result.responseText
                        + " uError : " + errorData);
                    alert("error! 다시 접속!");
                })
            })
        } else {
            alert("error! 다시 접속!");
        }
    })
});
```

1. 회원가입

usersSignUpPage.jsp



5. 개발일정 6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

usersController

```
//아파치와이자 게임정보 수정처리용
@RequestMapping(value = "/users/mypage_InfoForm.da", method = {RequestMethod.POST, RequestMethod.GET})
public String moveMypage_InfoForm(Users users, Model model){

    if(users.getUsers_password().trim() != null && users.getUsers_password().trim().length() > 0){
        if(this.bCryptPasswordEncoder.matches(users.getUsers_password().trim(), users.getUsers_password())){
            users.setUsers_password(this.bCryptPasswordEncoder.encode(users.getUsers_password().trim()));
        }else{
            users.setUsers_password(users.getUsers_password());
        }
    }

    if(usersService.updateUsers(users) > 0){
        return "common/main";
    }else{
        model.addAttribute("attributeValue", users.getUsers_email() + " 해당 이메일은 회원정보 수정이 불가능합니다.");
        return "users/main";
    }
}else{
    users.setUsers_password(usersService.selectUsers(users.getUsers_email()).getUsers_password());
    if(usersService.updateUsers(users) > 0){
        return "common/main";
    }else{
        model.addAttribute("attributeValue", users.getUsers_email() + " 해당 이메일은 회원정보 수정이 불가능합니다.");
        return "users/error";
    }
}
}
```



usersMapper

2. 마이페이지

usersMypge_InfoForn.jsp

개인정보 수정

- [개인정보 수정](#)
- [레시피 조회](#)
- [주문정보 조회](#)
- [탈퇴하기](#)

Mail(ID)	lbyk@gmail.com	
비밀번호	*****	
제입력	*****	
이름	김수진	
전화번호	01056455127	
우편번호	12419	검색
주소	경기도 아주시 아름로 120번길 8-1 (종점)	
개인정보 수정하기 홈페이지 돌아가기		



usersController

```
//마이페이지 정보조회 +수정작성
@RequestMapping(value = "/usersMypge_InfoForn.ds", method = {RequestMethod.POST, RequestMethod.GET})
public String moveMypge_InfoForn(users users, Model model){

    if(users.getUsers_password().trim() != null && users.getUsers_password().trim().length() > 0){
        if(this.bCryptPasswordEncoder.matches(users.getUsers_password().trim(), users.getUsers_password())){
            users.setUsers_password(this.bCryptPasswordEncoder.encode(users.getUsers_password().trim()));
        }else{
            users.setUsers_password(users.getUsers_password());
        }
    }

    if(usersService.updateUsers(users) > 0){
        return "common/main";
    }else{
        model.addAttribute("attribute", users.getUsers_email() + "님을 이용하시는 회원정보 수정이 불가능합니다.");
        return "common/main";
    }
}

users.setUsers_password(usersService.selectUsers(users.getUsers_email()).getUsers_password());
if(usersService.updateUsers(users) > 0){
    return "common/main";
}else{
    model.addAttribute("attribute", users.getUsers_email() + "회원 이용하시는 회원정보 수정이 불가능합니다.");
    return "users/main";
}
```

usersMapper

```
<!-- ID를 통해 회원정보 불러오기 -->
<select id="selectUsers" parameterType="string" resultMap="resultUsers" >
    select *
    from users
    where users_email = #{users_email}
</select>

<!-- 마이페이지 개인정보 수정 -->
<update id="updateUsers" parameterType="Users">
    update users
    set users_password = #{users_password},
        modified_at = sysdate,
        users_post_code = #{users_post_code},
        users_address1 = #{users_address1},
        users_address2 = #{users_address2}
    where users_email = #{users_email}
</update>
```

3. 회원탈퇴

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

usersMypage_DeleteForm.jsp

탈퇴하기

회원탈퇴를 진행하려면 본인의 ID/PWD를 입력해주세요

Mail(ID)

ID

이름

이름

전화번호

전화번호

돌보하기

홈페이지도 돌아가기

```
body
  div class="box">
    div align="center" style="text-align:center;">
      h3 style="margin:0;">회원가입
      form action="memberpage_updateform.do" method="post">
        div class="form-group">
          label style="margin-bottom:0.5em;">이름
          input type="text" class="form-control" name="users_name" id="users_name" placeholder="이름" style="width:100%;"/>
        
```

 div class="form-group">
 label style="margin-bottom:0.5em;">이메일
 input type="text" class="form-control" name="users_email" id="users_email" placeholder="이메일" style="width:100%;"/>

 div class="form-group">
 label style="margin-bottom:0.5em;">비밀번호
 input type="password" class="form-control" name="users_password" id="users_password" placeholder="비밀번호" style="width:100%;"/>

 div class="form-group">
 label style="margin-bottom:0.5em;">전화번호
 input type="text" class="form-control" name="users_phone" id="users_phone" placeholder="전화번호" style="width:100%;"/>

 div style="text-align:center; margin-top:10px;">
 button class="btn btn-default" type="submit" value="회원가입">회원가입
 button class="btn btn-default" type="button" value="취소">취소

```
//删除用户  
@RequestMapping(value = "/userpage_DeleteForm.ds", method = {RequestMethod.POST, RequestMethod.GET})  
public String moveUsersPage_DeleteForm(HttpServletRequest userpage_DeleteForm){}
```

```

// 다른 곳에서 사용하는 경우는 아래와 같이 사용합니다.
// @RequestMapping(value = "/users/username_DeleteForm.do", method = {RequestMethod.POST, RequestMethod.GET})
// public String deletedealers(@users users, Model model) {
//     Set result = usersService.deletedealers(users);
// }

if(result > 0){
    return "redirect:usersLogout.do";
}
else{
    model.addAttribute("users", users.getUsers_email());
    return "summaUsers";
}
}

```

usersMapper

```
<!-- 회원탈퇴 시 회원삭제 -->
<delete id="deleteUsers" parameterType="string">
    delete from users
    where users_email = #{users_email}
    and users_name = #{users_name}
    and users_phone = #{users_phone}
</delete>
```

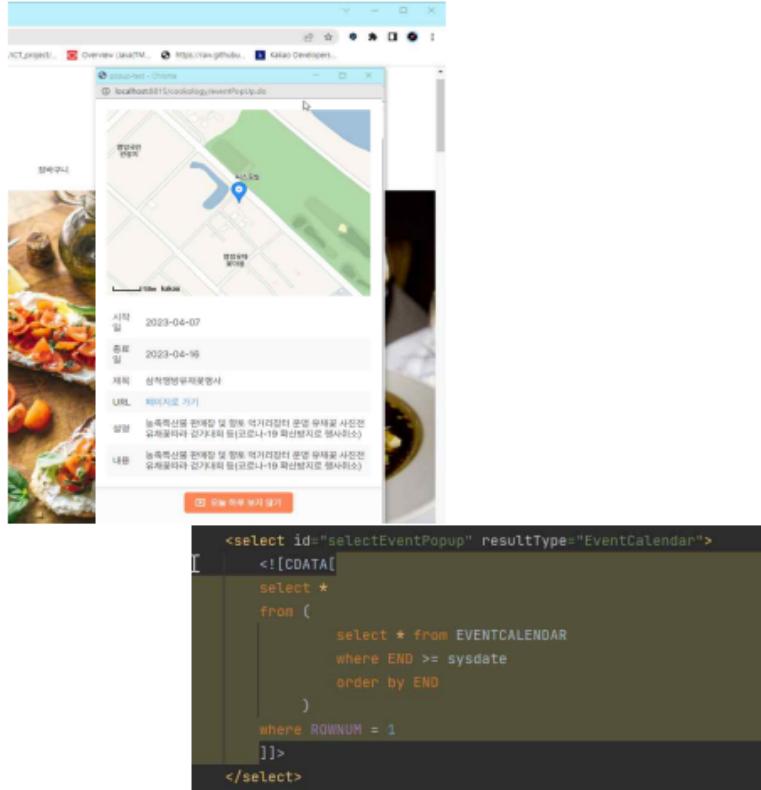
7. 주요 기능 / 코드

7-2. 김동준 : 이벤트페이지

Main Features and Codes



이벤트 팝업



5. 개발일정 6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
<@page contentType="text/html; charset=UTF-8" language="java" %>
<html>
<head>
</head>
<body>
<div id="popup" style="display: none;">
<div class="container">
<div>
<button class="event-button" onclick="hidePopupForToday()">
    <i class="far fa-window-close"></i> 오늘 해제 보기
</button>
</div>
</div>
</div>
</body>
</html>

<script>
    let eventDetail;
    $(document).ready(() => {...});

    document.addEventListener("DOMContentLoaded", function () {
        ...

        Function checkPopupVisibility() {...}

        function hidePopupForToday() {
            const now = new Date();
            const tomorrow = new Date(now.getFullYear(), now.getMonth() + 1, now.getDate());
            localStorage.setItem("hidePopupUntil", tomorrow.toISOString());
            document.getElementById("popup").style.display = "none";
            console.log("popup");
            window.close();
        }
    });
</script>

```

```

@RequestMapping(value = @Value("eventPopupDetail.dg"), method = RequestMethod.POST)
@ResponseBody
public String eventPopup() throws UnsupportedEncodingException {
    logger.info("eventPopup.dg : ");
    JSONObject jsonobjetc = new JSONObject();
    EventCalendar eventCalendar = eventService.selectPopupEvent();

    jsonobjetc.put("eventcalendar_uuid", eventCalendar.getEventcalendar_uuid().toString());
    jsonobjetc.put("start", eventCalendar.getStart().toString());
    jsonobjetc.put("end", eventCalendar.getEnd().toString());
    jsonobjetc.put("title", URLEncoder.encode(eventCalendar.getTitle()).toString(), encodingUTF8);
    jsonobjetc.put("url", eventCalendar.getUrl().toString());
    jsonobjetc.put("backgroundcolor", eventCalendar.getBackgroundcolor().toString());
    jsonobjetc.put("description", URLEncoder.encode(eventCalendar.getDescription()).toString(), encodingUTF8);
    jsonobjetc.put("content", URLEncoder.encode(eventCalendar.getContent()).toString(), encodingUTF8);
    jsonobjetc.put("longitude", eventCalendar.getLongitude().toString());
    jsonobjetc.put("latitude", eventCalendar.getLatitude().toString());

    return jsonobjetc.toJSONString();
}

```

Ntop , 전광판

위 : [USER01]

USER-POINT-TOP5

Rank	NAME	Points
1	user01	1460
2	user02	1340
3	user03	1245
4	user04	1210
5	user05	1186

RECIPE-VIEW-TOP5

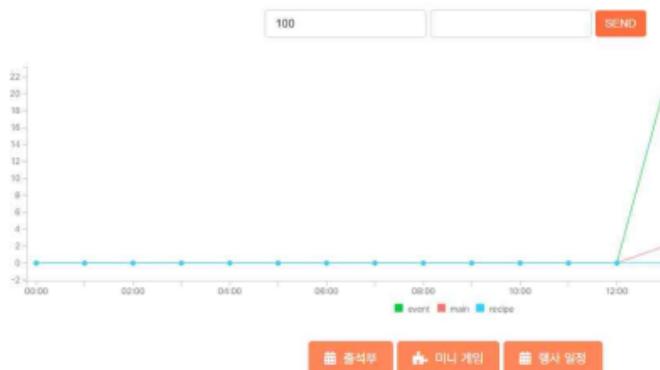
Rank	TITLE	VIEWS
1	t0e01	1460
2	t0e02	1340
3	t0e03	1245
4	t0e04	1210
5	t0e05	1186

RECIPE-GOOD-TOP

Rank	TITLE	GOOD
1	99e01	1460
2	99e02	1340
3	99e03	1245
4	99e04	1210
5	99e05	1186



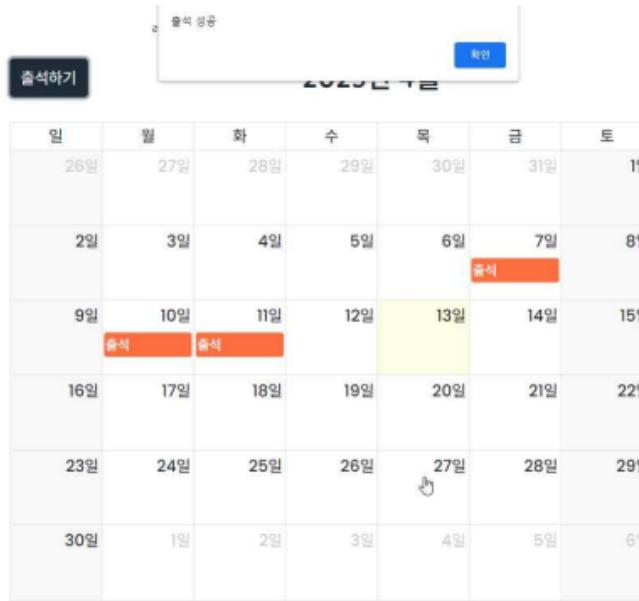
방문자수



```
■ danielkdj *
@ResponseBody
@RequestMapping(value = {"@v\"getVisitorLogsCount.do\""}, method = RequestMethod.POST)
public String getVisitorLogsCount() {
    List<VisitorLogsCount> list = eventService.selectVisitorLogs();
    JSONObject sendJson = new JSONObject();
    JSONArray jsonArray = new JSONArray();
    for (VisitorLogsCount visitorLogsCount : list) {
        JSONObject jsonObject = new JSONObject();
        jsonObject.put("page", visitorLogsCount.getPage().toString());
        jsonObject.put("count", String.valueOf(visitorLogsCount.getCount()));
        jsonArray.add(jsonObject);
    }
    sendJson.put("list", jsonArray);
    return sendJson.toJSONString();
}
```

```
<select id="selectVisitorLogs" resultType="VisitorLogsCount">
    <![CDATA[
        SELECT p.PAGE,
               h.hour,
               COUNT(v.TIME) AS count
        FROM (
            SELECT TRUNC(SYSDATE, '00') + (LEVEL - 1) / 24 AS hour
            FROM DUAL
            CONNECT BY LEVEL <= 24
        ) h
        CROSS JOIN (SELECT DISTINCT PAGE FROM VISITOR_LOGS) p
        LEFT JOIN VISITOR_LOGS v ON
            TRUNC(v.TIME, 'HH24') = h.hour AND v.PAGE = p.PAGE
        WHERE h.hour <= sysdate
        GROUP BY p.PAGE, h.hour
        ORDER BY p.PAGE, h.hour
    ]]>
</select>
```

출석부



The screenshot shows a web-based application for managing attendance. At the top, there's a header with the title '출석부' (Attendance). Below the header is a modal window with the title '출석 성공' (Attendance Success) and a blue '확인' (Confirm) button. The main area features a calendar grid for the month of August. The days are labeled from 26일 to 31일, followed by 1일. The days from 26 to 31 are light gray, while 1일 is white. A red box highlights the date '7일' (August 7th), which is labeled '출석' (Attendance). The days 10일 and 11일 are also highlighted in red, indicating they are also marked as attended.

일	월	화	수	목	금	토
26일	27일	28일	29일	30일	31일	1일
2일	3일	4일	5일	6일	7일	8일
9일	10일	11일	12일	13일	14일	15일
16일	17일	18일	19일	20일	21일	22일
23일	24일	25일	26일	27일	28일	29일
30일	1일	2일	3일	4일	5일	6일

```

I
function getAttendance() {
  $.ajax({
    url: "getAttendance.dal",
    type: "POST",
    data: {user_email: "#{user}"},
    dataType: "json",
    success: function(response) {
      console.log(`[requestPostBodyJson] : [response] : ${response}`);
      let jsonData = response;
      for (let i = 0; i < jsonData.list.length; i++) {
        let item = jsonData.list[i];
        let dateString = item.today.toString();
        let attendanceItem = {
          title: `${dateString}`,
          start: dateString,
          overlap: false,
          color: "#F0E68C"
        };
        attendanceMonthly.push(attendanceItem);
        calendar.addEvent(attendanceItem);
      }
      console.log(attendanceMonthly);
      calendar.render();
    },
    error: function(xhr) {
      console.log(`[requestPostBodyJson] : [error] : ${JSON.stringify(xhr)}`);
    },
  });
}
  
```

```

I
<select id="selectDetailEventCalendar" resultType="EventCalendar">
  select *
  from eventcalendar
  where eventcalendar_uuid = #{eventcalendar_uuid}
</select>
<select id="selectContinuousAttendance" resultType="int" ..>
<insert id="insertAttendance" parameterType="Attendance">
  insert into attendance(user_email,today) values (#{user_email},#{today})
</insert>
  
```

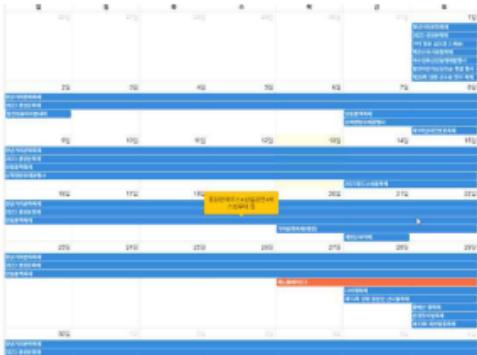
명언& 최대 출석일수

최대 연속 출석 횟수: 2



```
▲ danielkdj
@RequestMapping(value = {@"/continuousAttendance"}, method = {RequestMethod.POST})
@ResponseBody
public String continuousAttendance(@RequestParam String user_email) {
    JSONObject sendJson = new JSONObject();
    Integer day = null;
    try {
        day = this.eventService.selectContinuousAttendance(user_email);
        logger.info("con day: " + day);
    } catch (NullPointerException e) {
        logger.info("day is null");
        day = 0;
    } finally {
        logger.info(String.valueOf(day));
        sendJson.put("con", day.toString());
    }
    return sendJson.toJSONString();
}
```

이벤트 캘린더



```
@RequestMapping(value = @"/getEventCalendar.xls", method = RequestMethod.POST)
@ResponseBody
public String getEventCalendar() {
    ArrayList<EventCalendar> eventList = eventService.selectEventCalendar();
    Logger.info("eventCalendar_summary");
    JSONObject sendJson = new JSONObject();
    JSONArray jsonArray = new JSONArray();

    for (EventCalendar event : eventList) {
        JSONObject jsonObject = new JSONObject();
        jsonObject.put("eventcalendar_uuid", event.geteventcalendar_uuid().toString());
        jsonObject.put("title", event.getTitle().toString());
        jsonObject.put("start", event.getStart().toString());
        jsonObject.put("end", event.getEnd().toString());
        jsonObject.put("backgroundcolor", event.getBackgroundColor().toString());
        jsonObject.put("description", event.getDescription().toString());
        jsonArray.add(jsonObject);
    }

    sendJson.put("list", jsonArray);
    Logger.info(sendJson.toJSONString());
    return sendJson.toJSONString();
}
```

캘린더 디테일 뷰



시작일	2023-04-14
종료일	2023-04-16
제목	2023 청도소싸움축제
URL	페이지로 가기
설명	전통소싸움대회 및 공연행사
내용	전통소싸움대회 및 공연행사

```
@RequestMapping(value = @"/getEventCalendarDetail.do", method = {RequestMethod.POST, RequestMethod.GET})
@ResponseBody
public String getEventCalendarDetail(@RequestParam Map<String, Object> data) throws UnsupportedEncodingException {
    logger.info("event-detail");
    logger.info((String) data.get("uuid"));

    EventCalendar eventList = eventService.selectDetailEventCalendar((String) data.get("uuid"));

    JSONObject sendJson = new JSONObject();
    JSONArray jsonArray = new JSONArray();

    JSONObject jsonObject = new JSONObject();
    jsonObject.put("eventcalendar_uuid", eventList.getEventcalendar_uuid().toString());
    jsonObject.put("start", eventList.getStart().toString());
    jsonObject.put("end", eventList.getEnd().toString());
    jsonObject.put("title", URLEncoder.encode(eventList.getTitle().toString(), enc: "UTF-8"));
    jsonObject.put("url", eventList.getUrl().toString());
    jsonObject.put("backgroundcolor", eventList.getBackgroundColor().toString());
    jsonObject.put("description", URLEncoder.encode(eventList.getDescription().toString(), enc: "UTF-8"));
    jsonObject.put("content", URLEncoder.encode(eventList.getContent().toString(), enc: "UTF-8"));
    jsonObject.put("longitude", eventList.getLongitude().toString());
    jsonObject.put("latitude", eventList.getLatitude().toString());

    return jsonObject.toJSONString();
}
```

```
<select id="selectDetailEventCalendar" resultType="EventCalendar">
    select *
    from eventcalendar
    where eventcalendar_uuid = #{eventcalendar_uuid}
</select>
```

이벤트 추가

시작일: 연도-월-일

종료일: 연도-월-일

제목:

URL:

배경색:

설명:

```
@RequestMapping(value = "/addCalendarEvent.do", method = RequestMethod.POST)
@ResponseBody
public String addCalendarEvent(EventCalendar eventCalendar) {
    logger.info("addCalendarEvent.do : ");
    logger.info(eventCalendar.getContent());
    JSONObject jsonObject = new JSONObject();
    String uuid = UUID.randomUUID().toString();
    eventCalendar.setEventcalendar_uuid(uuid);

    if (eventService.insertEventCalendar(eventCalendar) > 0) {
        // 데이터 삽입 성공
        logger.info("insert success");
        jsonObject.put("status", "200");
        return jsonObject.toString();
    } else {
        // 데이터 삽입 실패
        jsonObject.put("status", "500");
    }
    return jsonObject.toJSONString();
}
```

```
<insert id="insertEventCalendar" parameterType="EventCalendar">
    insert into EventCalendar
    values
    (
        #{eventcalendar_uuid},#{start},#{end},#{title},#{url},
        #{backgroundcolor},#{description},#{content},#{longitude},#{latitude}
    )
</insert>
```

7. 주요 기능 / 코드

7-3. 정준혁 : 가맹점주(밀키트판매자) 페이지 , 밀키트 판매 페이지, KakaoMap API 를 이용한 '내 주변 식당'

Main Features and Codes

1. 상품 관리

5. 개발일정

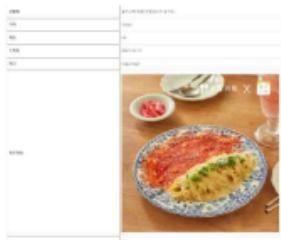
6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

상품 등록 페이지

사업자 번호	1010101010
상품명	풀지 다락 마른 크림파스타 알카트
상품금액	13900
재고	46
태그	tag2, tag3
썸네일	<input type="file"/> 793b5d4acb8_3ba73[1].jpg
내 용	2인분 난이도 중 15분 냉장보관
목록	<input type="button" value="등록하기"/>



1.1

이미지:

01[3].jpg

포토

설명글:

1.2

상품 등록

정보를 적고 등록을 누를 시 썸네일 이미지 사진과 정보가 저장됩니다.

등록한 정보는 유저가 접속 할 수 있는 밀키트 상품 홈페이지에서 볼 수 있게 됩니다.

대표 썸네일 상품에 대해서 밀키트 조리 이미지를 여러 개 등록 할 수 있습니다.

연속적으로 이미지를 등록한 후 확인을 하게 되면 오른쪽 이미지와 같이 조리 이미지 여러 개가 출력됩니다.



1.1 : JSP - productWriteForm.jsp

```
58 <cc:import url="/WEB-INF/views/common/header.jsp" />
59 <hr>
60 <h2 align="center">상품 등록 폼페이지</h2>
61 <br>
62 <form action="pinsert.do" method="post" enctype="multipart/form-data">
63   <table align="center" width="500" border="1" cellspacing="0">
64     <tr>
65       <td>
66         <th>상점주소</th>
67         <td><input name="business_number" value="1010101010" readonly></td>
68       </td>
69     </tr>
70     <tr>
71       <th>상품명</th>
72       <td><input type="text" name="product_name"></td>
73     </tr>
74     <tr>
75       <th>상품가격</th>
76       <td><input type="number" name="product_price"></td>
77     </tr>
78     <tr>
79       <th>재고</th>
80       <td><input type="number" name="stock"></td>
81     </tr>
82     <tr>
83       <th>태그</th>
84       <td><input type="text" name="tag"></td>
85     </tr>
86     <tr>
87       <th>첨부파일</th>
88       <td><input type="file" name="upfile"></td>
89     </tr>
90     <tr>
91       <th>내용</th>
92       <td><textarea name="content" rows="5" cols="50"></textarea></td>
93     </tr>
94     <tr>
95       <th colspan="2">
96         <button onclick="javascript:history.go(-1); return false;">취소</button>
97         &nbsp; <input type="submit" value="등록하기" style="margin-left: 10px;"> &nbsp;
98         <input type="reset" value="초기화" style="margin-left: 10px;">
99       </th>
100      </tr>
101    </table>
102  </form>
103  <br>
104  <br>
105 <cc:import url="/WEB-INF/views/common/footer.jsp" />
106 </body>
107 </html>
```

상품의 정보, 썸네일 파일을 등록하는 JSP입니다.

파일을 등록하기 위해 form 태그에 enctype을 사용합니다.



1.1 Controller - Product

등록 처리용 method

```
@RequestMapping(value = "pinsert.do", method = RequestMethod.POST)
public String productInsertMethod(Product product, Model model, HttpServletRequest request,
@RequestPart(name = "upfile", required = false) MultipartFile upfile) {
  String savePath = request.getSession().getServletContext().getRealPath("resources/img/product-img");
  if (upfile.isEmpty()) {
    String fileName = upfile.getOriginalFilename();
    if (fileName != null && fileName.length() > 0) {
      String renameFileName = FileNameChange.change(fileName, "yyyyMMddHHmmss");
      try {
        upfile.transferTo(new File(savePath + "\\") + renameFileName);
      } catch (Exception e) {
        e.printStackTrace();
        model.addAttribute("message", "파일이 등록되지 않았습니다!");
        return "common/error";
      }
      product.setOriginal_filepath(fileName);
      product.setRename_filepath(renameFileName);
    }
  }
  if (productService.insertProduct(product) > 0) {
    return "redirect:productList.do";
  } else {
    model.addAttribute("message", "실패!");
    return "common/error";
  }
}
```

공지사항 첨부파일 저장 폴더 경로를 지정하고

덮어쓰기 되는 것을 막기 위해, 파일명을 변경해서 폴더에 저장하는 방식을 쓰고 있습니다.

정말 온 파일 이름을 추출 한 후 바꿀 파일명에 대한 문자열을 만든 후 폴더에 저장 처리하는
method 입니다.

1.1 Mapper - Product

```
135<insert id="insertProduct" parameterType="Product">
136    insert into product (product_seq_id, business_number, thumb_url, product_name,
137                      product_price, stock, claims_num, CREATE_AT, modified_at, admin_inspection, tag,
138                      original_filepath, rename_filepath, content)
139    values ( SEQ_PRODUCT.nextval, #{business_number}, #{ thumb_url },
140              #{ product_name }, #{ product_price }, #{ stock }, #{ claims_num }, sysdate, sysdate, DEFAULT,
141              #{ tag }, #{ original_filepath, jdbcType=VARCHAR }, #{ rename_filepath, jdbcType=VARCHAR },
142              #{ content })
143 </insert>
```

Product 필드에 있는 값을 입력 받은 값을 insert(삽입)하는 query 문입니다. (상품 등록 시 사용)

1.2 : JSP - pDescriptionForm.jsp

```
117=<body>
118  <::import url="/WEB-INF/views/common/header.jsp" />
119  <br>
120= <script type="text/javascript"
121   src="${pageContext.servletContext.contextPath }/resources/js/jquery-3.6.3.win.js"></script>
122= <script>
123   <form action="ppinsert.do" method="post" id="my-form" enctype="multipart/form-data">
124   <div id="area-1">
125     <input type="hidden" id="product_seq_id" name="product_seq_id" value="${ product.product_seq_id }"/>
126   <script type="text/javascript">
127     const urlParams = new URLSearchParams(window.location.search);
128     const product_seq_id = urlParams.get('product_seq_id');
129     console.log(product_seq_id);
130     $('#product_seq_id').val(product_seq_id);
131   </script>
132   <label for="image">이미지:</label>
133   <input type="file" id="image" name="dupfile">
134   <br>
135   <label for="text">설명:</label>
136   <textarea id="text" name="description" rows="5" cols="50"></textarea>
137   <br>
138   </div>
139   <input type="submit" value="저장"/> &nbsp;
140   <input type="reset" value="취소"/>
141   <button onclick="javascript:history.go(-1); return false;">닫기</button>
142=</form>
143   <br>
144   <br>
145  <::import url="/WEB-INF/views/common/footer.jsp" />
146 </body>
147 </html>
```

한 상품에 대한 여러 개의 조리 이미지, 텍스트를 등록 하기 위한 jsp입니다.

dtail.do 요청시 상품등록할때의 값들과 상품에 대한 이미지들에 대한 값을

productMerchantDetailView.jsp에서 출력하게됩니다.

1.2 : Controller – ProductDescription

상세보기 method

```
107
108= @RequestMapping("pdetail.do")
109  public ModelAndView productDetailMethod(@RequestParam int product_seq_id) {
110    ModelAndView mav = new ModelAndView("product/productMerchantDetailView");
111
112    Product product = productService.selectProduct(product_seq_id);
113    ArrayList<ProductDescription> dlist = productDescriptionService.selectAll(product_seq_id);
114
115    if (product != null) {
116      mav.addObject("product", product);
117    }
118    if (dlist != null) {
119      mav.addObject("dlist", dlist);
120    }
121    return mav;
122  }
123
124  @RequestMapping(value = "ppinsert.do", method = RequestMethod.POST)
125  public String productDescriptionInsertMethod(ProductDescription productDescription, Model model,
126   HttpServletRequest request, @RequestParam(name = "product_seq_id") int product_seq_id,
127   @RequestParam(name = "description") String description,
128   @RequestParam(name = "dupfile", required = false) MultipartFile myfile) { // 파일업로드
129
130    // 파일이 있는경우 파일 경로
131    String savepath = request.getSession().getServletContext().getRealPath("resources/img/p-description-img");
132
133    productDescription = new ProductDescription();
134
135    productDescription.setProduct_seq_id(product_seq_id);
136    productDescription.setDescription(description);
137
138    logger.info("savePath = " + savepath);
139    logger.info("filename = " + myfile.getName());
140    if (!myfile.isEmpty()) {
141      logger.info("file = " + myfile);
142      String filename = myfile.getOriginalFilename();
143
144      logger.info(filename + "!");
145      if (filename != null && filename.length() > 0) {
146        String renamerfilename = filename.substring(0, filename.length() - 4) + "yyyyMMddHHmmss";
147        logger.info("new 파일명은 : " + filename + " , " + renamerfilename);
148        // 파일이 있는경우
149        try {
150          myfile.transferTo(new File(savepath + "\\" + renamerfilename));
151        } catch (Exception e) {
152          e.printStackTrace();
153          model.addAttribute("message", "첨부파일 저장 실패!");
154          return "common/error";
155        }
156
157        productDescription.setOriginal_img(filename);
158        productDescription.setRename_img(renamerfilename);
159      }
160    }
161
162    if (productDescriptionService.insertProductDescription(productDescription) > 0) {
163      return "redirect:productList.do";
164    } else {
165      model.addAttribute("message", "상세등록 실패!");
166      return "common/error";
167    }
168  }
```

1.2 : Mapper – ProductDescription

```
9<resultMap id="ProductDescription" type="ProductDescription">
10<result column="product_seq_id" property="product_seq_id"/>
11<result column="original_img" property="original_img"/>
12<result column="rename_img" property="rename_img"/>
13<result column="description" property="description"/>
14<result column="product_description_no" property="product_description_no"/>
15</resultMap>
16
20
21<insert id="insertProductDescription" parameterType="ProductDescription">
22    insert into product_description (product_seq_id, ORIGINAL_IMG, RENAME_IMG, DESCRIPTION, product_description_no)
23    values (
24        #{product_seq_id},
25        #{original_img,jdbcType=VARCHAR},
26        #{rename_img,jdbcType=VARCHAR},
27        #{description},
28        (select count(*) + 1
29         from PRODUCT_DESCRIPTION
30         where product_seq_id = #{product_seq_id})
31    )
32</insert>
33
```

한 상품에 대한 여러 개의 이미지, 텍스트를 insert 하는 query 문입니다.
여기서의 product_seq_id는 증가하지 않고, 다른 테이블에 있는 1씩 자동 증가하는
product_seq_id의 번호를 참조합니다.

2. 가맹점주 매출

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

Home > merchant > order management

매출 관리

검색할 등록날짜를 입력하세요: mm/dd/yyyy - mm/dd/yyyy

The total price is: 14000

The total price is: 14000

일매출: 14000 월매출: 14000

주문번호	상품명	상품 총 금액	유저이메일	취소여부	주문성공여부	주문시간
8	melon	2000	cyye.doe@example.com	N	Y	2023/04/09
7	melon	5000	ayye.doe@example.com	N	Y	2023/04/09
6	melon	4000	kyye.doe@example.com	N	Y	2023/04/09
5	melon	3000	aasde.doe@example.com	N	Y	2023/04/09
4	apple	2400	ssde.doe@example.com	N	Y	2023/04/09
3	apple	1600	jane.doe@example.com	N	Y	2023/04/09

2.1 : JSP - merchantOrderList.jsp

```

69=<div class="col-12 col-md-6 col-lg-4">
70=    <div class="single_catagory wow fadeInUp" data-wow-delay=".9s">
71=        
72=        <div class="catagory-title">
73=            <a href="${ pageContext.servletContext.contextPath }/merOrderList.do">
74=                <h5>매출관리</h5>
75=            </a>
76=        </div>
77=    </div>
78=    </div>
79=    </div>
80=    </div>
81= </section>
o>

<p>The total price is: ${ordersd.amount_day}</p>
<p>The total price is: ${ordersm.amount_month}</p>
<p>일매출: ${ordersd.amount_day}</p>
<p>월매출: ${ordersm.amount_month}</p>

<table>
<thead>
<tr>
<th>주문번호</th>
<th>상품명</th>
<!-- <th>상품가격</th> -->
<th>상품 품질</th>
<th>유저별매출</th>
<!-- <th>판매원별매출</th> -->
<th>판소리부</th>
<th>주문성공여부</th>
<th>주문시간</th>
</tr>
</thead>
<tbody>
<!-- 주문번호 -->
<c:forEach items="${ requestScope.list }" var="k">
<tr>
<td>${ k.orders_seq_id }</td>
<!-- 상품명 -->
<td>${ k.product_name }</td>
<!-- 상품 품질 -->
<td>${ k.amount }</td>
<!-- 유저별매출 -->
<td>${ k.user_email }</td>
<!-- 판소리부 -->
<td>${ k.cancellation }</td>
<!-- 주문 성공 여부 -->
<td>${ k.orders_success }</td>
<!-- 주문시간 -->
<td><fmt:formatDate value="${ k.create_at }" pattern="yyyy-MM-dd HH:mm" /><!-- yyyy-MM-dd HH:mm --></td>
</tr>
</c:forEach>
</tbody>
</table>

```

왼쪽 상단의 코드는 일매출, 월매출을 표시합니다.

오른쪽의 사진은 주문번호~주문시간까지 리스트를 표시합니다

2.1 : Controller – Orders & Mapper

```
78  
79 @RequestMapping("merOrderList.do") //모델뷰에  
80 public ModelAndView orderListMethod() { //pManagementPage.jsp  
81     ModelAndView mav = new ModelAndView("orders/merchantOrderList");  
82     ArrayList<Orders> list = ordersService.selectAll();  
83     Orders ordersd = ordersService.selectOrders();  
84     Orders ordersm = ordersService.selectMonthOrders();  
85  
86     if(ordersd != null || ((CharSequence) ordersd).isEmpty()) {  
87         mav.addObject("ordersd", ordersd);  
88     }  
89     if(ordersm != null || ((CharSequence) ordersm).isEmpty()){  
90         mav.addObject("ordersm", ordersm);  
91     }  
92     if(list != null) {  
93         mav.addObject("list", list);  
94     }else {  
95         mav.addObject("message",  
96                     "상품 정보가 존재하지 않습니다.");  
97     }  
98     return mav;  
99 }  
100  
101 }
```

<!-- 상품 리스트 전체 목록 -->

```
<select id="selectAll" resultMap="resultMerchant">  
    select * from orders  
    order by orders_seq_id desc  
</select>
```

<!-- 일 매출 구하기-->

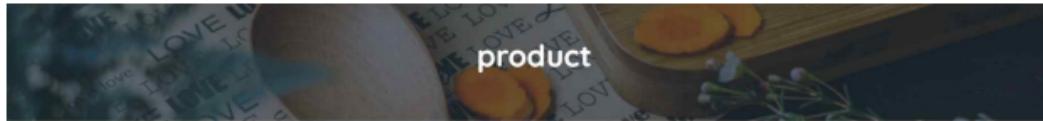
```
<select id="selectOrders" resultMap="resultMerchant">  
    select *  
    from (  
        SELECT BUSINESS_NUMBER, SUM(amount) as amount_day  
        FROM orders  
        WHERE TRUNC(create_at) = TRUNC(create_at)  
        AND business_number = '1010101010'  
        GROUP BY business_number  
    )  
</select>
```

<!-- 월 매출 구하기-->

```
<select id="selectMonthOrders" resultMap="resultMerchant">  
    <![CDATA[  
        select *  
        from (  
            SELECT BUSINESS_NUMBER, SUM(amount) as amount_month  
            FROM orders  
            WHERE create_at >= TRUNC(sysdate, 'MONTH')  
            AND create_at < ADD_MONTHS(TRUNC(sysdate, 'MONTH'), 1)  
            AND business_number = '1010101010'  
            GROUP BY BUSINESS_NUMBER  
        )]]>  
</select>
```

날짜(date)를 원하는 단위(여기서는 'MONTH' 즉, 월)까지 자릅니다.

3. 밀키트 상품



Home > product

간단한 정보들이 가맹점주가 상품을 등록할 때 이 페이지에 추가되어 집니다.

"기승전풀 김창전골 밀키트" 29900 원 상품 보기	"을지다락 매콤 크림 파스타 밀 키트" 12900 원 상품 보기	"채선당 월 남쌈 밀키 트" 16900 원 상품 보기	"채선당 샤 브샤브 밀 키트" 15900 원 상품 보기	"채선당 밀 푀유나베 밀 키트" 16900 원 상품 보기	"바게트 피 자와 토마 토 스파게 티 밀키트" 14900 원 상품 보기
---	--	---	--	---	--

Home > product > product detail
Start Bootstrap HOME ABOUT SHOP Contact

을지다락 매콤 크림파스 타 밀키트



39900원 13900원

2인분

단인도 품

제조

3단계

설정

포장지갑 낱장/여크막스(기후 변화에 따라 스트로폼 박스로 대체)

배송장과 낱장 낱장을 분리해놓은 시장 중

유통기간 제조일로부터 7일

중량 550g

1 Add to cart.



3.1 : JSP - merchantOrderList.jsp

```

<section class="py-5">
  <div class="container px-4 px-lg-5 mt-5">
    <div class="row gx-4 gx-lg-5 row-cols-2 row-cols-md-3 row-cols-xl-4 justify-content-center">
      <c:forEach items="${ requestScope.plist }" var="p">
        <div class="col mb-5">
          <div class="card h-100">
            <!-- Product image-->
            <!-- 사이즈, 맞춤기 -->
            
            <!-- Product details-->
            <div class="card-body p-4">
              <div class="text-center">
                <!-- Product name-->
                <h5 class="fw-bolder">${ p.product_name }</h5>
                <!-- Product price-->
                ${ p.product_price } 원
              </div>
            </div>
            <!-- Product actions-->
            <div class="card-footer p-4 pt-0 border-top-0 bg-transparent">
              <c:url var="uProduct" value="userProductDetail.do">
                <c:param name="product_seq_id" value="${ p.product_seq_id }" />
                <%--   <c:param name="product_seq_id" value="${ product_seq_id }" /> --%>
              </c:url>
              <div class="text-center">
                <a class="btn btn-outline-dark mt-auto" href="${ uProduct }">상품
                  보기</a>
              </div>
            </div>
          </div>
        </div>
      </c:forEach>
    </div>
  </div>
</section>

```

유저가 보게 되는 상품 리스트 기능입니다.

위에 부분은 밀키트 상품 리스트에서 보여지는 쌤네일, 일부 정보들입니다.

아래부분은 밀키트 상세 상품에서 보여지는 쌤네일,정보, 밀키트 조리 이미지, 조리 이미지 설명이 들어가게 됩니다

3.1 : Controller –ProductDescription & Mapper

```
//유저상품상세페이지로 이동
@RequestMapping(value = "userProductDetail.do", method = { RequestMethod.GET, RequestMethod.POST })
public ModelAndView moveKakaoMapView(@RequestParam int product_seq_id) {
    ModelAndView mav = new ModelAndView("product/userDetailProduct");

    Product product = productService.selectProduct(product_seq_id);
    ArrayList<ProductDescription> dlist = productDescriptionService.selectAll(product_seq_id);

    logger.info("pdlist222", dlist);
    logger.info("pdlist333" + dlist);

    if (product != null) {
        mav.addObject("product", product);
    }
    if (dlist != null) {
        mav.addObject("dlist", dlist);
    }
    return mav;
}
```

```
<!-- 상품 상세보기 요청 처리용 , 상품 수정 -->
<select id="selectProduct" resultMap="resultProduct" parameterType="_int">
    select * from product
    where product_seq_id = #{ product_seq_id }
</select>
```

```
<!-- 상품 리스트 전체 목록 -->
<select id="selectAll" resultMap="resultProduct">
    select * from product
    order by product_seq_id desc
</select>
```

7. 주요 기능 / 코드

7-4. 전병현 : 고객센터 페이지(QNA, FAQ, 공지사항)

Main Features and Codes

공지사항 기능

```
작업 사용자 : byunghyun
@RequestMapping(value=@"/ntop5.do", method=RequestMethod.POST)
@ResponseBody
public String noticeNewTop5Method()
    throws UnsupportedEncodingException {
    // ajax 요청이, 이런 방법은 어리석기가 있음
    // response 객체 이용
    // 1. 출첵스팀으로 응답하는 방법 (아이디 중복 체크 예)
    // 2. 뷰리풀버로 리턴하는 방법 : response body에 값을 저장함

    // 최근 공지를 5개 조회해 옵
    ArrayList<Notice> list = noticeService.selectNewTop5();
    logger.info("ntop5.do : " + list.size()); // 5 출력 확인

    // 전송용 json 객체 준비
    JSONObject sendJson = new JSONObject();
    // 리스트 저장할 json 배열 객체 준비
    JSONArray jarr = new JSONArray();

    // list 를 jarr 에 옮기기 (복사)
    for(Notice notice : list) {
        // notice 의 각 풀드값 저장할 json 객체 생성함
        JSONObject job = new JSONObject();

        job.put("noticeno", notice.getNoticeno());
        // 한글에 대해서는 인코딩해서 json에 담도록 함

        // 한글제정 방지
        job.put("noticetitle", URLEncoder.encode(notice.getNoticetitle(), enc("utf-8")));
        // 날짜는 반드시 toString() 으로 문자열로 바꿔서
        job.put("noticecontent", URLEncoder.encode(notice.getNoticecontent(), enc("utf-8")));
        // json에 담아야 함
        job.put("noticedate", notice.getNoticedate().toString());

        jarr.add(job); // job 을 jarr에 추가함
    }
}
```

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css"
      crossorigin="anonymous"/>
<script type="text/javascript">
$(function () {
    var values = $('#newnotice').html();
    console.log(values);
    $.ajax({
        url: "ntop5.do",
        type: "post",
        dataType: "json",
        success: function (data) {
            console.log("success : " + data);
            const jsonStr = JSON.stringify(data);
            const json = JSON.parse(jsonStr);

            for(let i in json.list) {
                values += '<div class="panel panel-default" data-wow-delay=".1.' + (i*2) + 's">';
                + '<div class="panel-heading">';
                + '<h4 class="panel-title">';
                + '<a class="accordion-toggle collapsed" data-toggle="'
                + '(collapse)' data-parent="#accordion" href="#collapse'+ i +'>';
                + decodeURIComponent(json.list[i].noticetitle).replace(/#/gi, " ") + '</a>';
                + '<small>' + json.list[i].noticedate + '</small>';
                + '</h4></div>';
                + '<div id="collapse'+ i +'" class="panel-collapse collapse">';
                + '<div class="panel-body">';
                + decodeURIComponent(json.list[i].noticecontent).replace(/#/gi, " ");
                + '</div></div></div>';
            }
        }
    });
});
```



By Marian | May 19, 2017 10 12
The 10 Best Bars By The Seaside In Blackpool, UK

Tiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation

[Continue Reading...](#)



By Marian | May 19, 2017 10 12

How To Get A Narcissist To Feel Empathy

Tiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation

[Continue Reading...](#)



By Marian | May 19, 2017 10 12

Bristol's 10 Amazing Weekend Brunches And Late Breakfasts

Tiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation

[Continue Reading...](#)



By Marian | May 19, 2017 10 12

10 Of The Best Places To Eat In Liverpool, England

공지 서비스 오픈10
2023-05-24
공지 서비스가 오픈되었습니다. 많이 이용해 주세요.



공지 서비스 오픈9
2023-05-24
공지 서비스가 오픈되었습니다. 많이 이용해 주세요.

공지 서비스 오픈8
2023-05-24

공지 서비스 오픈7
2023-05-24
공지 서비스가 오픈되었습니다. 많이 이용해 주세요.

공지 서비스 오픈6
2023-05-24

```

0-제의 사용위치 : byunghyun

@RequestMapping(value="@*\"nsearchTitle.do", method={RequestMethod.POST, RequestMethod.GET})
public String noticeSearchTitleMethod(
    @RequestParam("keyword") String keyword,
    @RequestParam("page") int page,
    Model model) {

    int limit = 10;
    int listCount = noticeService.selectSearchTitleCount(keyword);

    Paging paging = new Paging(listCount, page, limit, keyword);
    paging.calculator();

    logger.info("paging"+ paging);
    ArrayList<Notice> list =
        noticeService.selectSearchTitle(paging);

    if(list != null && list.size() > 0) {
        model.addAttribute( attributeName: "list", list);
        model.addAttribute( attributeName: "url",   attributeValue: "nsearchTitle.do");
        model.addAttribute( attributeName: "paging", paging);
        return "notice/noticePage";
    }else {
        model.addAttribute( attributeName: "list", list);
        return "notice/noticePage";
    }
}

```

```

<script>

$(function () {
    $('.panel-heading a').on('click', function () {
        var currentPanel = $(this).attr('href');
        var currentPanelState = $(currentPanel).hasClass('in');

        if (currentPanelState) {
            $('.panel-collapse').collapse('hide');
            return;
        }

        $('.panel-collapse').collapse('hide');
        setTimeout(function () {
            $(currentPanel).collapse('show');
        }, 150);
    });
});

function search() {
    var search = document.getElementById("searchBar").value;
    var searchType = "";
    if (document.getElementById("noticeAllRadio").checked) {
        searchType = "list";
    } else if (document.getElementById("noticeContentRadio").checked) {
        searchType = "nsearchContent";
    } else if (document.getElementById("noticeTitleRadio").checked) {
        searchType = "nsearchTitle";
    }
    location.href = searchType+"?do?keyWord=" + search+ "&page=1";
}

```

질문하기(Q&A)

공지사항

NOTICE search	
<input checked="" type="radio"/> All	<input type="radio"/> Content
<input type="radio"/> Title	<input type="button" value="search"/>
123	2023-04-05
2235	2023-04-05
글자 서비스가 오른쪽되었습니다. 많이 이용해 주세요.	
글자 서비스 오른쪽	2023-03-30

금지사항

The screenshot shows the KakaoTalk search interface. At the top, there's a search bar with the placeholder text "검색어를 입력하세요...". Below the search bar, a red button labeled "검색하기" (Search) is visible. The search results are displayed in a grid format. The first result is titled "NOTICE search" with a subtitle "125". The second result is titled "2238". Each result card has a "공지사항" (Notice) icon and two buttons at the bottom right: "공지수정하기" (Edit notice) and "삭제" (Delete). The overall background is white with orange highlights on the search bar and result cards.



제시미 펠리트 이벤트 배송조회 장바구니

공지 하기

四

五

留待



자주하는 질문(FAQ)

질문하기(Q&A)

공지사항

NOTICE search

123
2020-01-01

123
2020-01-01

질문 기능

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
0개의 사용위치 * byunghyun *
@RequestMapping("QNAPage.do")
public String moveQNAPage(@RequestParam("user_email") String user_email, Model model) {
    user_email = "admin@example.com";
    logger.info("user_email : " + user_email);

    try { //로그인 한 경우
        Users users = usersService.selectUsers(user_email);
        if (users.getIs_admin().equals("Y")) {
            logger.info("QNA forwarded");
            ArrayList<QNA> admin_list = QNAservice.selectNewQuestionList();
            try { //QNA가 있을 경우
                model.addAttribute( attributeName: "list", admin_list);
                return "userService/QNAPage";
            } catch (Exception e) { //QNA가 없을 경우
                model.addAttribute( attributeName: "list", attributeValue: "");
                return "userService/QNAPage";
            }
        } else { //관리자 아닌 경우
            ArrayList<QNA> user_list = QNAservice.selectMyQNAList(user_email);
            try { //QNA가 있을 경우
                model.addAttribute( attributeName: "list", user_list);
                return "userService/QNAPage";
            } catch (Exception e) { //QNA가 없을 경우
                model.addAttribute( attributeName: "list", attributeValue: "");
                return "userService/QNAPage";
            }
        }
    } catch (Exception e) { //로그인 안한 경우
        return "users/loginPage";
    }
}
```

```
<c:if test="${ loginUser.is_admin != 'Y' }">
    <c:url var="QuestionPage" value="QuestionPage.do">
        <c:param name="user_email" value="${ loginUser.user_email }" />
    </c:url>
    <button class="event-button" onclick="javascript:location.href='${ QuestionPage }';">
        <i class="fas fa-question-circle"></i> 질문하기
    </button>
</c:if>
<br/>

<div class="panel-group" id="accordion">
    <c:if test="${ loginUser.is_admin == 'Y' }">
        <div class="faqHeader">답변 안된 질문내역 조회 </div>
    </c:if>
    <c:if test="${ loginUser.is_admin != 'Y' }">
        <div class="faqHeader">나의 질문내역보기</div>
    </c:if>
    <c:set var="list" value="${list}" />
```



Q&A

자주하는 질문(FAQ)

질문하기(QNA)

공지사항

자주하는 질문과 관련된 내용입니다. 자주하는 질문은 고객님의 편의를 위해 정리해놓았습니다.

FAQ

Q&A

별점 주관식에게 있는 좋은 별점 주관해주세요.



자주하는 질문(FAQ)

질문하기(QNA)

공지사항

자주하는 질문과 관련된 내용입니다. 이전에서 연결하여 편리한 정보를 이용해주세요.

답변 안된 질문내역조회

집에서 쉽게 만드는 디저트집에서 쉽게 만들 수 있는 디저트 레시피 추천해주세요.

피자 반죽 레시피집에서 만들 수 있는 피자 반죽 레시피 알려주세요.

Q&A답변하기

초콜릿 틀 주기 만들기초콜릿 틀 주기를 만드는데 필요한 재료와 과정은 무엇인가요?

222444444555

레시피 딜리브 이벤트 배송조회 청바구니

자주하는 질문(FAQ)

질문하기(QNA)

공지사항

자주하는 질문과 관련된 내용입니다. 이전에서 연결하여 편리한 정보를 이용해주세요.

질문하기

나의 질문내역보기

별점 주관서울에 있는 좋은 별점 주관해주세요.

dksefhfd2023-04-05 21:57:31

답변 안된 질문에 답변하기

번호 *

user6@example.com

질문: 피자 반죽 레시피

집에서 만들 수 있는 피자 반죽 레시피 알려주세요.

답변 *

수정하기

자주하는 질문 기능

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

Cookology

FAQ

자주하는 질문(FAQ)
질문하기(QNA)
공지사항

FAQ

200
20
500
30
고기를 요리할 때 안전한 온도는 뭘까요?
신선한 허브는 어떻게 보관해야 하나요?

Lookology

FAQ 수정하기

번호 *

2

질문 *

20

답변 *

500

● 수정하기

FAQ

200
20
500
30
고기를 요리할 때 안전한 온도는 뭘까요?
신선한 허브는 어떻게 보관해야 하나요?

자주하는 질문(FAQ)
질문하기(QNA)
공지사항

FAQ

20
20
500
300
고기 올루에 햄을 만든은 대부분 온도가 다릅니다. 소고기, 돼지고기, 쑥부지고기, 햄(고기는 56°F (13°C), 모든 가공육은 145°F (63°C)입니다.

7. 주요 기능 / 코드

7-5. 윤영광 : 주문, 장바구니, 주문 및 결제, 배송조회 페이지

Main Features and Codes

1. 장바구니 추가하기



클라우드 스테이크 밀키트(2인)

19900원

5
Add to cart

장바구니

상품 이미지	상품 이름	가격	수량
	[맛집키트] 짐승파스타 월타감버스 암 아히요 밀키트(2-3인)	16900	1
	클라우드 스테이크 밀키트(2인)	19900	2
	클라우드 스테이크 밀키트(2인)	19900	5
총 가격:			0

주문하기

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

장바구니

상품 이미지	상품 이름	가격	수량
	[맛집키트] 짐승파스타 월타감버스 암 아히요 밀키트(2-3인)	16900	1
	클라우드 스테이크 밀키트(2인)	19900	2
	클라우드 스테이크 밀키트(2인)	19900	5
총 가격:			16900

주문하기

장바구니

상품 이미지	상품 이름	가격	수량
	[맛집키트] 짐승파스타 월타감버스 암 아히요 밀키트(2-3인)	16900	1
	클라우드 스테이크 밀키트(2인)	19900	2
	클라우드 스테이크 밀키트(2인)	19900	5
총 가격:			16400

주문하기

1. 장바구니에 상품 추가하고 이동하기

```
@Autowired  
private CartService cartService;  
  
@Autowired  
private ProductService productService;  
  
// 상품을 장바구니에 추가하기  
  
@RequestMapping(value=@-"cartInsert.do", method={RequestMethod.GET, RequestMethod.POST})  
public ModelAndView addToCart(@RequestParam("product_seq_id") int product_seq_id [Complexity is 5 Everything is cool]  
        , @RequestParam("amount") int amount  
        , HttpSession session) {  
  
    // 1. 현재 세션에 저장된 사용자 정보를 가져옴  
    Users loginUsers = (Users) session.getAttribute( < "loginUsers");  
  
    // 2. 로그인되어 있지 않은 경우, 로그인 페이지로 이동  
    if (loginUsers == null) {  
        return new ModelAndView( viewName: "redirect:/loginPage.do");  
    }  
  
    // 3. 로그인 되었을 경우 상품을 장바구니에 추가  
    cartService.addToCart(loginUsers.getUser_email(), product_seq_id, amount);  
  
    // 4. 상품 추가 후 장바구니 페이지로 이동  
    return new ModelAndView( viewName: "redirect:/cartList.do");  
}
```

상품을 장바구니에 추가하는 addToCart Method

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
46 // 장바구니 페이지로 이동하기  
47 @RequestMapping(value=@-"cartList.do" , method={ RequestMethod.GET , RequestMethod.POST })  
48 @@@@ public ModelAndView moveCartListPage(HttpServletRequest session) { Complexity is 7 It's time to do something.  
49  
50  
51 // 1. 로그인된 사용자인지 세크 : 현재 세션에 저장된 사용자 정보를 가져옴  
52 Users loginUsers = (Users) session.getAttribute( < "loginUsers");  
53  
54  
55 // 2. 로그인되어 있지 않은 경우, 로그인 페이지로 이동  
56 if (loginUsers == null) {  
57     return new ModelAndView( viewName: "redirect:/loginPage.do");  
58 }  
59  
60  
61 // 3. 로그인된 사용자의 장바구니 목록을 가져오기 : 사용자의 이메일로 장바구니 목록을 가져옴  
62 String user_email = loginUsers.getUser_email();  
63  
64 ArrayList<Cart> cartList = cartService.getCartListWithProductInfo(user_email);  
65  
66 // 4. 각 장바구니 항목에 대한 상품 정보 가져오기  
67 HashMap<Integer, Product> productList = new HashMap<>();  
68 for (Cart cart : cartList) {  
69     int product_seq_id = cart.getProduct_seq_id();  
70     Product product = productService.selectProduct(product_seq_id);  
71     productList.put(product_seq_id, product);  
72 }  
73  
74 List<String> selectedProductNames = new ArrayList<>();  
75 for (Cart cart : cartList) {  
76     int product_seq_id = cart.getProduct_seq_id();  
77     Product product = productService.selectProduct(product_seq_id);  
78     selectedProductNames.add(product.getProduct_name());  
79 }
```

장바구니 페이지로 이동하는 moveCartListPage Method

```
// 장바구니에 등록된 상품 갯수 수정하기
@RequestMapping(value=@Value("updateCartItem.do", method={RequestMethod.GET, RequestMethod.POST})
public ModelAndView updateCartItem(@RequestParam("cart_seq_id") int cart_seq_id
    , @RequestParam("amount") int amount, HttpSession session) {
    cartService.updateCartItem(cart_seq_id, amount);

    // 장바구니 페이지로 이동
    return new ModelAndView( viewName: "redirect:/cartList.do");
}

// 장바구니에 등록된 상품 삭제하기
@RequestMapping(value=@Value("removeCartItem.do", method={RequestMethod.GET, RequestMethod.POST})
public ModelAndView removeCartItem(@RequestParam("cart_seq_id") int cart_seq_id, HttpSession session) {

    // 3. 장바구니에 있는 상품을 제거함
    cartService.removeCartItem(cart_seq_id);

    // 상품 삭제 후 장바구니 페이지로 이동
    return new ModelAndView( viewName: "redirect:/cartList.do");
}
```

장바구니에 등록된 상품 갯수 수정하는 updateCartItem
, 장바구니에 등록된 상품 삭제하는removeCartItem Method

3. cart-mapper.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE mapper PUBLIC
  "-//mybatis.org//DTD Mapper 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="cartMapper">

  <resultMap type="Cart" id="resultCart">
    <id property="cart_seq_id" column="cart_seq_id" />
    <result property="user_email" column="user_email" />
    <result property="product_seq_id" column="product_seq_id" />
    <result property="price" column="price" />
    <result property="amount" column="amount" />
  </resultMap>  Yun, 2023-04-11 오후 18:47 • 2023.04.11 신규작업 권도.

  <resultMap id="resultCartWithProductInfo" type="Cart" extends="resultCart">
    <association property="product" javaType="Product">
      <id column="product_seq_id" property="product_seq_id" />
      <result column="product_name" property="product_name" />
      <result column="product_price" property="product_price" />
      <result column="rename_filepath" property="rename_filepath" />
    </association>
  </resultMap>

  <!-- 카트구니에 상품 추가 (Add product to cart) -->
  <insert id="addToCart" parameterType="Cart">
    INSERT INTO CART (CART_SEQ_ID, USER_EMAIL, PRODUCT_SEQ_ID, AMOUNT, PRICE)
    VALUES (cart_seq_id.NEXTVAL, #{user_email}, #{product_seq_id}, #{amount}, #{price})
  </insert>
```

5. 개발일정 6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
<!-- 카트구니에 상품 추가 (Add product to cart) -->
<insert id="addToCart" parameterType="Cart">
  INSERT INTO CART (CART_SEQ_ID, USER_EMAIL, PRODUCT_SEQ_ID, AMOUNT, PRICE)
  VALUES (cart_seq_id.NEXTVAL, #{user_email}, #{product_seq_id}, #{amount}, #{price})
</insert>

<!-- 로그인한 사용자의 카트 목록 조회 (Retrieve the cart list of the logged-in user) -->
<select id="getCartList" resultType="Cart">
  SELECT * FROM CART WHERE USER_EMAIL = #{user_email}
</select>

<!-- 카트내 상품 수량 수정 (Update the quantity of a product in the cart) -->
<update id="updateCartItem">
  UPDATE CART SET AMOUNT = #{amount} WHERE CART_SEQ_ID = #{cart_seq_id}
</update>

<!-- 카트내 상품 삭제 (Delete a product from the cart) -->
<delete id="removeCartItem">
  DELETE FROM CART WHERE CART_SEQ_ID = #{cart_seq_id}
</delete>

<select id="getCartListWithProductInfo" resultMap="resultCartWithProductInfo">
  SELECT c.* , p.product_name, p.product_price, p.rename_filepath
  FROM cart c
    INNER JOIN product p ON c.product_seq_id = p.product_seq_id
    WHERE c.user_email = #{user_email}
</select>

</mapper>
```

cart-mapper.xml 파일 쿼리문

4. 장바구니 view 페이지

5. 개발일정 6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
<table class="cart-table">
    <thead>
        <tr>
            <th></th>
            <th>상품 이미지</th>
            <th>상품 이름</th>
            <th>가격</th>
            <th>수량</th>
        </tr>
    </thead>
    <tbody>
        <c:forEach items="${cartList}" var="cartItem">
            <tr>
                <td>
                    <input type="checkbox" name="product_seq_id" value="${cartItem.product_seq_id}" data-price="${cartItem.price * cartItem.amount}" onclick="updateTotalPrice()" />
                </td>
                <td>
                    
                </td>
                <td>${productList[cartItem.product_seq_id].product_name}</td>
                <td style="..." ${cartItem.price}</td>
                <td style="..." ${cartItem.amount}</td>
            </tr>
        </c:forEach>

        /* --- 총 가격을 계산하는 부분 --- */
        <c:set var="totalPrice" value="0" />
        <c:forEach items="${cartList}" var="cart">
            <c:if test="${fn:contains(param.selectedItems, cart.product_seq_id)}">
                <c:set var="totalPrice" value="${totalPrice + (cart.amount * cart.price)}" />
            </c:if>
        </c:forEach>
    
```

5. 장바구니 view 페이지

5. 개발일정 6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
// 채크된 상품의 총 가격 계산하는 함수
function updateTotalPrice() {
    var selectedItems = getSelectedItems();
    var totalAmount = selectedItems.reduce(function (accumulator, item) {
        return accumulator + item.price;
    }, 0);
    document.getElementById("totalPrice").innerHTML = totalAmount;
}

// 결제포트 초기화
var IMP = window.IMP;
IMP.init("*****");

// 채크된 상품 목록을 가져오는 함수
function getSelectedItems() {
    var checkboxes = document.getElementsByName("product_seq_id");
    var selectedItems = [];

    for (var i = 0; i < checkboxes.length; i++) {
        if (checkboxes[i].checked) {
            var item = {
                product_seq_id: checkboxes[i].value,
                price: parseInt(checkboxes[i].getAttribute("data-price"), 10)
            };
            selectedItems.push(item);
        }
    }
    return selectedItems;
}
```

```
// 결제 모듈 함수
function requestPay() {
    var selectedItems = getSelectedItems();
    var today = new Date();
    var hours = today.getHours();
    var minutes = today.getMinutes();
    var seconds = today.getSeconds();
    var milliseconds = today.getMilliseconds();
    var MerchantUid = hours + minutes + seconds + milliseconds;
    // 선택한 상품이 없으면 경고 메시지 표시
    if (selectedItems.length === 0) {
        alert("장바구니에서 상품을 선택하세요.");
        return;
    }

    // 선택한 상품의 총 가격 계산
    var totalAmount = selectedItems.reduce(function (accumulator, item) {
        return accumulator + item.price;
    }, 0);
    IMP.request_pay({
        pg: "html5_inicis",
        pay_method: "card",
        merchant_uid: MerchantUid,
        name: "${productNamelist}",
        amount: totalAmount,
        buyer_email: "${buyerEmail}",
        buyer_name: "${buyerName}",
        buyer_tel: "${buyerTel}",
        buyer_addr: "${buyerAddr1} ${buyerAddr2}",
        buyer_postcode: "${buyerPostcode}"
    }, function (rsp) {
        if (rsp.success) {
            alert("결제가 완료되었습니다.");
            // 결제 성공시 글씨색상
        } else {
    }
```

cartList.jsp script 태그 관련 내용

1. 체크박스 체크한 상품들 가격 * 수량 계산해서 실시간으로 총가격란에 입력하기.

2. Import API 이용해서 결제하기

6. 배송조회용 tracking.jsp

```
<div class="form-group">
    <input type="hidden" id="selectedCompanyName" name="selectedCompanyName" value="" />
    <select id="shippingCompanySelect" name="shippingCompanyName" onchange="setSelectedCompanyName();">
        <option value="" disabled selected>
            고객님의 택배 회사를 선택하세요
        </option>
        <option value="01">우체국택배</option>
        <option value="04">한국화물운송</option>
        <option value="05">한진택배</option>
        <option value="06">롯데택배</option>
        <option value="07">롯데택배</option>
        <option value="11">풀로지스틱</option>
        <option value="12">EMS</option>
        <option value="13">DHL</option>
        <option value="14">UPS</option>
        <option value="15">DTS(상사택배)</option>
        <option value="17">한국택배</option>
        <option value="18">한국택배</option>
        <option value="20">풀스</option>
        <option value="21">FedEx</option>
        <option value="22">대한택배</option>
        <option value="23">경기택배</option>
        <option value="24">GS 편의점 택배</option>
        <option value="25">TNT Express</option>
        <option value="26">DPS</option>
        <option value="27">GS칼텍스</option>
        <option value="28">한국택배</option>
        <option value="29">OML Robot Mail</option>
        <option value="30">Parcels</option>
        <option value="31">KBS</option>
        <option value="32">ECOS Express</option>
        <option value="40">2@4</option>
        <option value="41">GS Express</option>
        <option value="42">PST택배</option>
    </select>
```

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
<option value="123">한국택배</option>
<option value="124">LTL</option>
<option value="125">GTS로지스</option>
<option value="126">아울타코리아</option>
<option value="127">로지스넷</option>
<option value="128">만월도로지스틱</option>
<option value="129">홈플 오월드박</option>
<option value="130">UFC로지스</option>
<option value="131">글리래빗</option>
<option value="132">K포티</option>
<option value="134">에이케이홈딩스</option>
<option value="135">MTS</option>
<option value="136">케이제이티</option>
<option value="137">내비오</option>
<option value="138">리스트마일</option>
<option value="139">오플러스 러쉬</option>
<option value="140">직구문</option>
<option value="141">언티로지스</option>
<option value="142">생그먼고</option>
<option value="143">두데이</option>
<option value="144">큐브플로우(CUBEFLOW)</option>
<option value="145">현대글로벌</option>
<option value="146">KJ로지스틱(KSE)</option>
<option value="147">에스터쉬링</option>
<option value="148">ARBO</option>
<option value="149">골드스냅</option>
<option value="151">자마언트</option>
<option value="152">(주)한티인터넷에서님</option>
<option value="153">(주)이지로지스틱</option>
<option value="154">KT EXPRESS(케이티익스프레스)</option>
<option value="155">HY</option>
<option value="156">유파로지스</option>
<option value="157">우진언터로지스</option>
```

```
<script type="text/javascript">
    function setSelectedCompanyName() {
        const shippingCompanySelect = document.getElementById("shippingCompanySelect");
        const selectedCompanyName = shippingCompanySelect.options[shippingCompanySelect.selectedIndex];

        // Set the value of the hidden input field
        document.getElementById("selectedCompanyName").value = selectedCompanyName;

        // Return true to allow form submission
        return true;
    }

    // ...
    // ...
    // ...

    function numericOnly(input) {
        input.value = input.value.replace(/\D/g, "");
    }

    function validateForm() {
        const shippingCompanySelect = document.getElementById("shippingCompanySelect");
        const trackingNumber = document.getElementById("tracking_number").value;

        if (shippingCompanySelect.value === "") {
            alert("Please select a shipping company.");
            return false;
        }

        if (trackingNumber.trim() === "") {
            alert("Please enter a tracking number.");
            return false;
        }

        return true;
    }
}
```

7. 배송조회용 trackingSearch

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
// 배송조회 메이저로 이동  
@RequestMapping(value=@Value("trackingSearch.do", method={ RequestMethod.GET, RequestMethod.POST })  
public String moveLoginPage() {  
    return "order/trackingSearch";  
}  
  
// 배송조회 페이지로 이동  
@RequestMapping(value=@Value("trackingPage.do", method={RequestMethod.GET , RequestMethod.POST })  
public String moveTrackingPage() {  
    return "order/tracking";  
}
```



```
</style>  
</head>  
<body>  
    <div class="header-area"><a href="main.do">메인페이지로 이동</a></div>  
    <div class="title-content">  
        <div class="title-notice">운송장 번호</div>  
        <div class="title-invoice">${param.tracking_number}</div>  
        <div class="title-company">${param.selectedCompanyName}</div>  
    </div>
```



8. 배송조회용 trackingSearch

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

The screenshot illustrates the process of tracking a delivery using the Lookology application. It starts with a search bar at the top containing the text "trackingSearch". Below the search bar, there's a navigation menu with links like "최근 본 상품" (Recently Viewed Products) and "메인페이지로 이동" (Move to Main Page). The main content area features a large image of a candle and some flowers, with the text "배송조회" overlaid.

The user begins by selecting a delivery company from a dropdown menu labeled "고객님의 택배 회사를 선택해주세요" (Select your delivery company). The dropdown shows "경동택배" as the selected option. A red arrow points from this selection to the tracking number input field below.

The tracking number "2125151234" is entered into the input field. A red box highlights this tracking number, and another red arrow points from it to the "조회하기" (Search) button below.

After clicking "조회하기", the page transitions to a new view showing the tracking history. At the top right, there are icons for "문서함 번호" (Document Number), "2125151234", and "경동택배". The tracking history is listed on the right side of the screen:

- 서울 마포구 빅백로 23 | 배송완료
2023-04-14 12:30:35
- 서울 남대문 | 배송출발
2023-04-14 07:30:20
- 남서울터미널 | 배달전
2023-04-13 08:45:20
- 대전HUB | 간선상자
2023-04-13 08:57:10
- 성복직영 | 김화처리
2023-04-12 11:30:00
- 고객 | 인수차점록
2023-04-12 10:42:50

At the bottom of the tracking history, there's a note: "① 배송 정보 제공 : 스마트택배 API". The footer of the page features the "스마트택배" logo.

9. 배송조회용 trackingSearch

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

```
// 배송조회 페이지로 이동  
@RequestMapping(value=@Value("trackingSearch.do", method={ RequestMethod.GET, RequestMethod.POST })  
  
public String moveLoginPage() {  
    return "order/trackingSearch";  
}
```

```
// 배송조회 페이지로 이동  
@RequestMapping(value=@Value("trackingPage.do", method={RequestMethod.GET, RequestMethod.POST })  
  
public String moveTrackingPage() {  
    return "order/tracking";  
}
```

```
</style>  
</head>  
<body>  
    <div class="header-area"><a href="main.do">메인페이지로 이동</a></div>  
    <div class="title-content">  
        <div class="title-notice">운송장 번호</div>  
        <div class="title-invoice">${param.tracking_number}</div>  
        <div class="title-company">${param.selectedCompanyName}</div>  
    </div>
```

10. 결제하기

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

The image shows a composite screenshot of an e-commerce application's payment flow.

Top Left: A shopping cart summary titled "장바구니" (Cart). It lists three items: a salad dish, a 2-3 person meal kit, and two pieces of steak. The total amount is 116,400 원.

Middle Left: A red box highlights the "주문하기" (Order Now) button at the bottom of the cart page.

Top Right: A payment method selection screen for "KG 이니시스". It shows various payment options like Pay, KPay, and various cards. The total amount is 116,400 원.

Bottom Right: A payment confirmation screen from "KG 이니시스". It shows the transaction details: "주문금액 388,700 원" (Order Amount 388,700 won), payment method "내이자리 카드 결제" (Naejabari Card Payment), and a green "결제완료" (Payment Complete) button.

10. 결제하기

5. 개발일정

6. Usecase Diagram

7. 주요기능 / 코드

8. 마무리

The image shows a composite screenshot of an e-commerce application's payment flow.

Top Left: A shopping cart summary titled "장바구니" (Cart). It lists three items: a salad dish, a 2-in Steak Fillet, and a 5-in Steak Fillet. The total amount is 116,400 원.

Middle Left: A red box highlights the "주문하기" (Order Now) button at the bottom of the cart page.

Top Right: A payment method selection screen for "KG 이니시스". It shows various payment options like Pay, KPay, and various cards. The total amount is 116,400 원.

Bottom Right: A payment confirmation screen showing the transaction details. The total amount is 388,700 원. It includes a QR code for payment and a green "결제하기" (Pay) button.

후기

Review



정준혁 후기

1. 세미프로젝트 시작

ICT 인재 개발원에 다닌지 대략 3개월이 진행되었을 때 세미 프로젝트를 시작하게 되었다. 우리는 풀스택으로 백엔드와, 프론트엔드 둘 다 진행하게 되었다.

처음 접해보는 프로젝트였고 많이 미숙했던 나였지만 무엇을 할까 팀원들과 곰곰히 고민하다가 '레시피'라는 주제로 초점이 맞추어졌고, 만개의 레시피, 그리팅 몰이라는 웹페이지를 바탕으로 3주간 프로젝트를 진행하였다.

나는 여기서 '가맹점주'가 이용할 수 있는 기능들을 담당하였다.
전체적인 카데고리에서 뽑자면, 상품관리, 매출관리, 밀키트 상품에 대한 기능이다.

프로젝트를 시작하면서 수도 없이 느낀 건 팀원들과 마음이 맞는 게 중요한 점 중 하나라고 생각한다.
다행이 팀원간 불화도 없었고, 팀원들 모두 프로젝트에 대한 의욕이 활활 타올라서 이런저런 기능들, 프로그램도 사용하였다.
막히는 부분이 있으면 자신의 일도 멈춰 두고, 팀원들과 같이 나아가려는 모습과 프로젝트가 끝난 지금도 본인이 원래 프로젝트에 구현 하고 싶었던 기능들을 완성시키기 위해
이곳 저곳 물어보면서 완성시키려고 하는 열정적인 모습이 앞으로 취업 후에 대한 마음가짐으로 보여져서 너무 좋았다.

2. 아쉬운 점

어렵다. 시간이 부족하다? 이건 내가 초반부터 한 주 전까지 공부하면서 평계처럼 느꼈었던 감정이다.

팀원분중 한 분이 말씀 해주셨던게 생각난다. '나이가 이렇게 먹은만큼 남들보다 더 안 자야지'

나도 나름 만족스럽게 공부하고, 잠도 적게 잔다고 생각했지만 착각이었다.

코딩 실력이 미숙한 만큼. 나는 더 열심히 했어야 했고, 모르는게 있을때는 잠을 안자면서까지 이해하려고 노력 했어야했다.

서론이 길었는데, 본론을 말하자면.. 내게 아쉬운 점
spring framework mvc 모델링에 대해서 정확하게 이해하지 못했다.
깃헙에 대한 속지가 미숙했다.
ajax,css를 적절하게 활용하지 못했다.
정도가 될 것이다.

김동준 후기

프로젝트를 진행하면서 많은 것을 배울 수 있었고,
작업을 진행하면서 백업과 깃을 통한 프로젝트의 관리 필요성을
크게 느낄 수 있었고, 각 작업 파트를 마지막에 통합하는 과정에서
수많은 문제가 있었는데 애자일 방법론을 통한 유지보수 관리의
필요성도 느끼게 되었다.

김수진 후기

– 아쉽거나 반성해야 할 점

1. 첫번째 세미 프로젝트라는 점을 생각하지 않고 많은 기능을 구현하고자 함
2. 전체적으로 본게 아니라 한가지의 기능을 오래 붙잡고 있었던 점(시간분배 부족)
3. 백엔드를 중점으로 진행되었음(프로트엔드 부분이 많이 아쉬웠음)
4. 깃허브와 백업의 중요성에 대해서 배울 수 있었음

– 좋았다고 생각되는 점

1. 예상결과에 비해서는 기능들이 전부 구현되지는 않았지만 끝까지 프로젝트를 포기하지 않았다는 점
2. 이번 세미프로젝트를 바탕으로 이후의 프로젝트를 어떤 식으로 진행하면 될지 생각할 수 있게 되었다는 점

Thanks

한국 ICT 인재개발원

클라우드 환경의 자바 개발자 양성과정 A

세미 프로젝트 1팀 : OAOC

Email

윤영광 : greenbeatle12@gmail.com

김동준 : danieldj@naver.com

김수진 : itsyksj@gmail.com

전병현 : atill4584@gmail.com

정준혁 : yus9537@naver.com

조현진 :

2023.04.14



Thank you

Chef Presentation Template Design

