

종합설계 프로젝트 수행 보고서

프로젝트명	데이터 분석을 통한 데이트 장소 추천 시스템
팀번호	S4-8
문서제목	수행계획서(O) 2차발표 중간보고서(O) 3차발표 중간보고서(O) 최종결과보고서(O)

2020.12.04

팀원 : 김효연 (팀장)

팀원 : 김기원 (팀원)

지도교수 (인)

문서 수정 내역

작성일	대표작성자	버전(Revision)	수정내용	
2020.01.15	김효연(팀장)	1.0	수행계획서	최초 작성
2020.01.17	김효연(팀장)	1.1	수행계획서	개발내용 수정
2020.02.01	김효연(팀장)	2.0	수행계획서	상세설계 작성
2020.04.06	김효연(팀장)	2.1	수행보고서	모듈 수정
2020.04.30	김효연(팀장)	3.0	수행보고서	프로토타입 작성
2020.05.02	김효연(팀장)	3.1	수행보고서	사용함수 작성
2020.06.26	김효연(팀장)	4.0	수행보고서	코드 작성
2020.06.27	김효연(팀장)	4.1	수행보고서	사용함수 수정
2020.06.27	김효연(팀장)	4.2	수행보고서	데모화면 캡처
2020.12.04	김효연(팀장)	5.0	수행보고서	결론 추가

문서 구성

진행단계	프로젝트 계획서 발표	중간발표1 (2월)	중간발표2 (4월)	학기말발표 (6월)	최종발표 (10월)
기본양식	계획서 양식	계획서 양식	계획서 양식	계획서 양식	계획서 양식
포함되는 내용	I. 서론 (1~6) II. 본론 (1~3) 참고자료	I. 서론 (1~6) II. 본론 (1~4) 참고자료	I. 서론 (1~6) II. 본론 (1~5) 참고자료	I. 서론 (1~6) II. 본론 (1~7) 참고자료	I. 서론 (1~6) II. 본론 (1~7) III. 결론 (1~2) 참고자료

이 문서는 한국산업기술대학교 컴퓨터공학부의 “종합설계” 교과목에서
프로젝트

“데이터 분석을 통한 데이트 장소 추천 시스템”을 수행하는
(S4-8, 김효연, 김기원)들이 작성한 것으로 사용하기 위해서는 팀원들의
허락이 필요합니다.

목 차

I. 서론

1. 작품선정 배경 및 필요성
2. 기존 연구/기술동향 분석
3. 개발 목표
4. 팀 역할 분담
5. 개발 일정
6. 개발 환경

II. 본론

1. 개발 내용
2. 문제 및 해결방안
3. 시험시나리오
4. 상세 설계
5. Prototype 구현
6. 시험/ 테스트 결과
7. Coding & DEMO

III. 결론

1. 연구 결과
 2. 작품제작 소요재료 목록
- ※ 참고자료

I. 서론

1. 작품선정 배경 및 필요성

필요성	내 용
수요배경	<ul style="list-style-type: none">• 현대 사회에서 많은 데이터가 존재함에 따라 사람들이 결정에 대한 어려움을 겪음• 빅 데이터 시대에 도래 하면서 정부나 기업에서 방대한 양의 데이터를 활용하여 차별화된 데이터를 제공하는 추세
필요성 1	<ul style="list-style-type: none">• 다양한 데이터를 통한 객관적인 추천 데이터 표출• 시스템은 불필요한 자료를 필터링 할 수 있는 기능을 수행• 선택에 도움을 줄 수 있는 객관적 데이터를 제공
필요성 2	<ul style="list-style-type: none">• 사용자가 요구하는 특정 키워드를 바탕으로 한 장소 추천 시스템의 필요• 빅 데이터 분석 편의를 위한 데이터 사전 구현의 필요
필요성 3	<ul style="list-style-type: none">• 최신트렌드에 맞춰 개개인의 특성에 맞는 맞춤형 서비스를 제공할 필요

2. 기존 연구 / 기술동향 분석

개발내용	내 용
국내 관련연구 소개, 장점, 단점	<ul style="list-style-type: none"> ■ 모바일 상황인식 추천 맛집 서비스 개발 [1] <ul style="list-style-type: none"> • (소개)위치정보를 기반으로 맛집 서버에서 근처 맛집의 위치를 알려줌 • (장점)근처의 추천 맛집 정보 제공, 특정 맛집 후기 확인 가능 • (단점)사용자의 상황을 인식하여 개인 맞춤형 서비스를 제공하는 것의 어려움 ■ 빅데이터 기반 사용자 맞춤 헬스케어, 웰니스 추천 서비스 [2] <ul style="list-style-type: none"> • (소개)데이터를 데이터 마이닝으로 분류하여 사용자 맞춤형 의료정보를 제공하는 웰니스 정보 제공 서비스 • (장점)디테일 검색 기능을 강화하고 데이터 프록시 서버 기능을 도입하여 분석의 효율성이 극대화됨 • (단점)웹 사이트 정보 제공이므로 신뢰성이 부족하고 추가적인 분석의 정확성 확보가 필요 ■ 상품 평가 분석 및 사용자 구매 기준을 반영한 상품 추천 시스템 [3] <ul style="list-style-type: none"> • (소개)사용자들의 비정형화된 리뷰 데이터에 대해 의미 있는 정보를 생성하고, 정보를 사용자에게 추천해주는 시스템 • (장점)사용자들의 평가를 기반으로 한 다양하고 확실한 정보의 제공 • (단점)사전에 등록되지 않은 신조어에 대한 의미를 해석하고 반영하는 것에 대한 문제
국외 관련연구 특징, 장점, 단점	<ul style="list-style-type: none"> ■ Adaptive web search based on user profile constructed without any effort from users [4] <ul style="list-style-type: none"> • (소개)사용자의 검색 기록을 기반으로 용어 가중치를 계산하여 사용자 정보 요구에 적합한 검색 결과 필터링 • (장점)사용자의 선호도에 맞는 검색 시스템으로 검색어 추천 및 자동완성 기능 • (단점)방대한 사용자 검색기록에서 특정 사용자의 선호 검색어를 완벽하게 파악하기 어려움 ■ Context-aware intelligent recommendation system for tourism [5] <ul style="list-style-type: none"> • (소개)협업 필터링, 콘텐츠 기반 추천 및 인구 통계 학적 프로파일링으로 구성된 하이브리드 기반 추천시스템 • (장점)위치정보에 크게 의지하지 않는 추천정보 출력 • (단점)부적절한 콘텐츠 필터링 및 정보 과부하가 발생
공통적인 문제점	<ul style="list-style-type: none"> ■ 데이터 필터링 능력 부족 <ul style="list-style-type: none"> • 관리하는 데이터가 증가할수록 필요한 정보와 불필요한 정보를 구분하여 필터링 하는 것이 어려워지며, 정보 과부하가 발생 • 데이터에 대한 신뢰성이 떨어지며, 이에 대한 추가적인 분석으로 정확성을 높여줄 필요가 있음 ■ 개인을 위한 맞춤형 서비스 제공 문제 <ul style="list-style-type: none"> • 방대한 양의 데이터를 활용하는 것으로 사용자 개개인이 직접적으로 원하는 정보를 구분하여 서비스를 제공 해주는 것이 어려움 ■ 특정 단어를 바탕으로 한 검색 기능 부재 <ul style="list-style-type: none"> • 사용자가 원하는 자료를 찾기 어려움

3. 개발목표

최종목표

- 데이트 장소 추천 시스템을 통해 장소 결정에 대한 효율적인 선택을 제공

단계	내 용
1단계	<ul style="list-style-type: none"> ■ 크롤링을 통한 여러 장소에 대한 다양한 데이터 값 추출 • 웹상에서 데이터 추출을 위한 대상 선정 • 데이트 장소와 관련된 데이터 값 추출 후 DB에 저장
2단계	<ul style="list-style-type: none"> ■ 수집된 데이터를 시각화 • 저장된 데이터에 빈도수를 기준으로 한 키워드 추출 • 편집거리 알고리즘을 이용하여 특정 키워드와 연관된 하위 키워드의 도출 • 수집된 데이터를 통해서 추출한 키워드들을 시각적으로 표현
3단계	<ul style="list-style-type: none"> ■ 데이트 장소 추천 시스템 구현 • 웹 페이지 구현을 통한 사용자 접근성 향상 • 사용자로부터 입력받은 키워드를 통한 검색결과 도출 • 사용자간의 리뷰를 바탕으로 한 맞춤형 데이트 장소를 추천

4. 팀 역할 분담

이 름	학 번	연 락 처	역 할 분 담
김효연 (조장)	2014150011	010-4037-4448	<ul style="list-style-type: none"> • 프로젝트 총괄 및 계획 • 웹 사이트 구현 및 시각화 그래프 구현 • 장소 캘린더 기능 구현 • 서비스 테스트
김기원 (조원)	2014154003	010-4563-8116	<ul style="list-style-type: none"> • AWS Server 및 Database 구축, 연동 • 크롤링 프로그램 구현 및 데이터 수집 • 회원 데이터 관리 체계 구축 • 서비스 테스트

5. 개발 일정

개발 일정	12월	1월	2월	3월	4월	5월	6월	7월	8월	9월	10월
사전조사 및 제안서 발표	●										
자료수집 및 스터디	●	●	●								
시스템 설계		●	●								
구현			●	●	●	●					
데모 테스트				●	●	●	●				
문서화 및 발표								●	●		
산업기술대전									●	●	
최종보고서 작성 및 발표										●	●

6. 개발 환경

개발 환경	• Linux 18.04
개발 언어	• Python, MySQL, Django
사용 프레임 워크	• PyCharm, MySQL Workbench
서버 환경	• Amazon EC2 Linux AMI 2018.03.0(HVM)
주요 라이브러리	• BeautifulSoup, Selenium
데이터 베이스	• AWS RDS MySQL

II. 본론

1. 개발 내용

순번	개발 내용
1	<ul style="list-style-type: none"> ■ 데이터 수집을 위한 크롤링 프로그램 개발 <ul style="list-style-type: none"> • Python를 이용한 크롤링 프로그램 구축 • Instagram이나 Twitter같은 소셜미디어의 API를 대상으로 한 데이터 수집
2	<ul style="list-style-type: none"> ■ 데이터 저장을 위한 데이터베이스 환경 구축 <ul style="list-style-type: none"> • AWS RDS에 MySQL을 연동한 데이터베이스 환경을 구축
3	<ul style="list-style-type: none"> ■ 키워드 도출 시스템 구현 <ul style="list-style-type: none"> • 데이터 빈도 수로 키워드 추출 • 편집거리 알고리즘을 사용하여 키워드 간의 연결을 시각화
4	<ul style="list-style-type: none"> ■ 데이트 장소 추천 웹 구현 <ul style="list-style-type: none"> • 웹 서버에서 시각화된 데이터 값을 볼 수 있는 웹 환경을 구현 • AWS EC2를 사용하여 웹 서버 동작
5	<ul style="list-style-type: none"> ■ 회원 시스템을 통한 개인형 맞춤 서비스 구현 <ul style="list-style-type: none"> • 데이트 장소 간 평점 및 리뷰 작성 • 회원별 리뷰를 활용하여 유사도가 높은 회원 간 선호 장소 추천
6	<ul style="list-style-type: none"> ■ 조회 수를 통한 인기 장소 선정 <ul style="list-style-type: none"> • 조회 수가 높은 상위의 장소 정보를 한 눈에 확인할 수 있도록 출력
7	<ul style="list-style-type: none"> ■ 장소 캘린더 기능 구현 <ul style="list-style-type: none"> • 날씨, 계절과 같은 키워드를 바탕으로 한 추천 장소를 캘린더 형식으로 제공

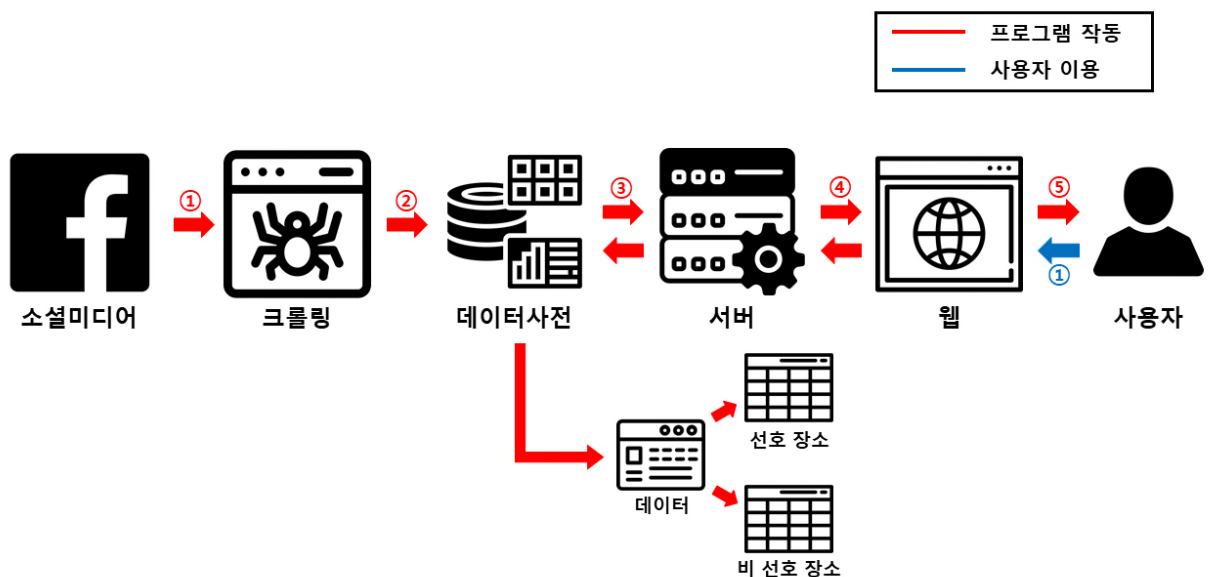
2. 문제 및 해결방안

문제점	해결 방법
도출된 결과에 대한 평가방법 부재	<ul style="list-style-type: none"> • 기존 검색결과와 협업필터링 알고리즘을 거친 추천 장소를 비교하여 평가
추천된 장소와 선호도를 측정할 수 있는 방법의 부재	<ul style="list-style-type: none"> • 장소 데이터별로 조회수를 카운트 하여 인기 장소를 선정 • 협업 필터링을 알고리즘을 사용하여 사용자간의 리뷰를 비교하여 선호 데이트 장소를 추천
관련 없는 데이터 수집 문제	<ul style="list-style-type: none"> • 크롤링 데이터 수집 시 불용어 처리와 예외 처리로 보다 직관적인 장소 데이터 수집
회원 맞춤형 서비스 제공 문제	<ul style="list-style-type: none"> • 사용자의 검색기록과 조회기록이 아닌 키워드 선택을 통한 키워드별 장소 데이터 제공

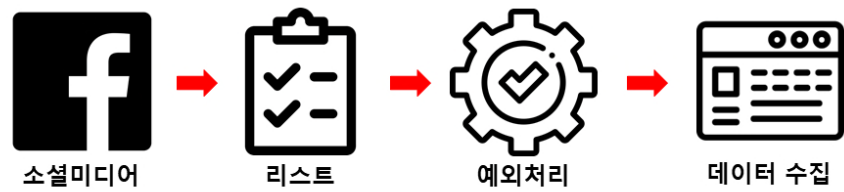
3. 시험시나리오

시나리오명	내 용
데이터 수집	<ul style="list-style-type: none"> Twitter, Instagram같은 소셜 미디어를 크롤링 하여 데이터를 수집
데이터 저장	<ul style="list-style-type: none"> 크롤링 대상 데이터를 데이터베이스에 저장 정제한 데이터 값을 데이터베이스에 저장
데이터 분석	<ul style="list-style-type: none"> 저장된 데이터를 분석하여 키워드 도출 도출된 키워드를 시각화
서버 전송	<ul style="list-style-type: none"> 데이터베이스에서 전달받은 값을 통해 웹으로 구현함
웹 서비스	<ul style="list-style-type: none"> 데이트 장소 추천 서비스 제공 인기 순위 리스트 출력 개인형 맞춤 서비스 장소캘린더

전체 시나리오



크롤링 시나리오



키워드 필터링 시나리오



개인형 맞춤 서비스 시나리오



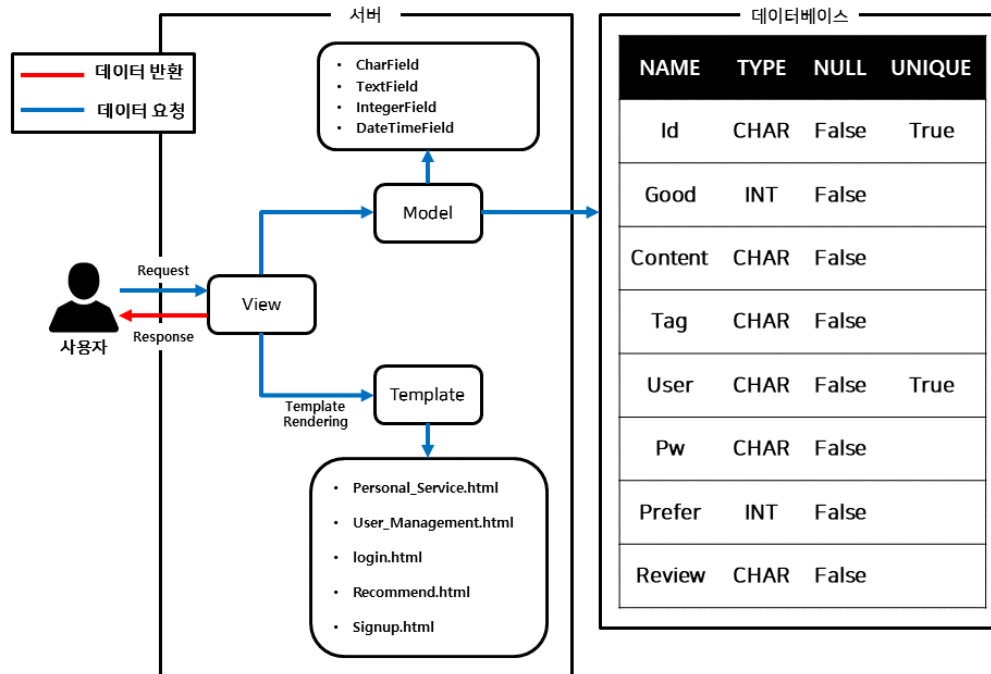
인기장소 선정 시나리오



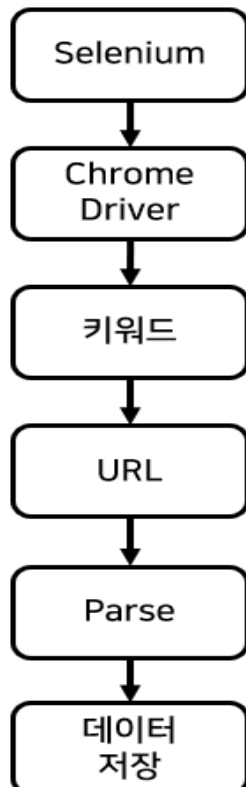


4. 상세 설계

시스템 구조 상세 설계

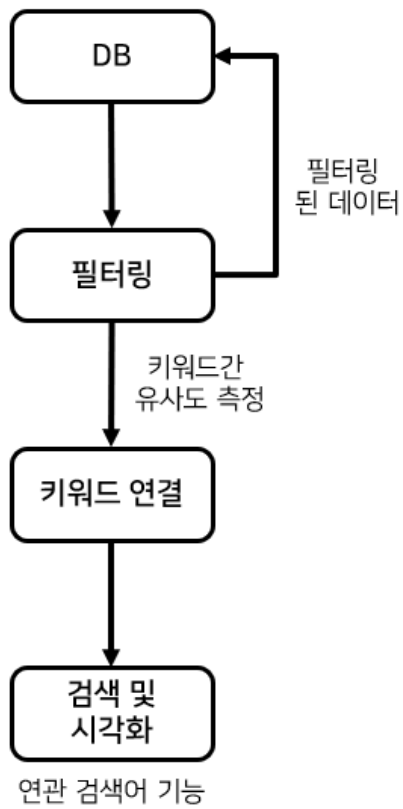


시스템 기능 모듈 상세설계(크롤링)

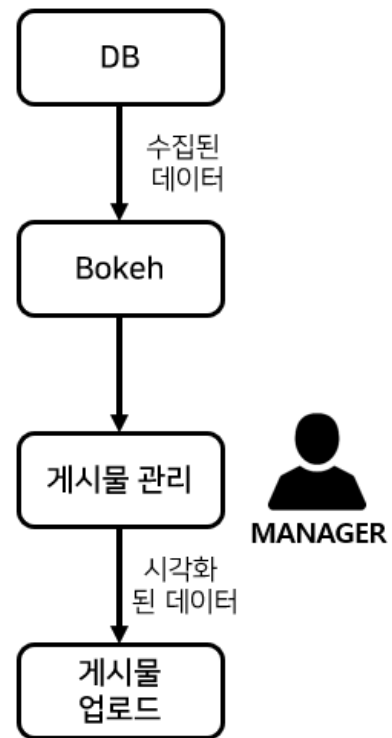


- Instagram 특성 상 url을 특정 지을 수 없기 때문에 chrome driver를 사용하여 게시물의 고유 url을 획득한다.
- 검색할 키워드를 입력한다.(ex.데이트 장소, 맛집)
- 게시물에서 크롤링 할 데이터를 추출한다.
 Parse1 = 'id'
 Parse2 = 'good'
 Parse3 = 'content'
 Parse4 = 'tag'

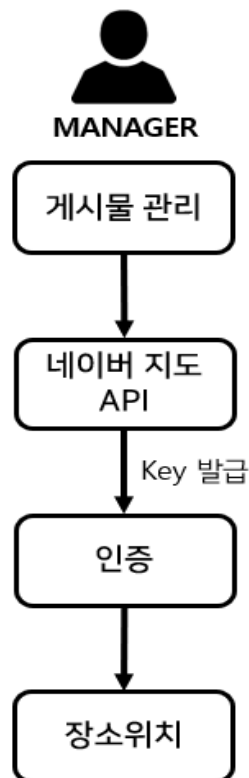
시스템 기능 모듈 상세설계(키워드)



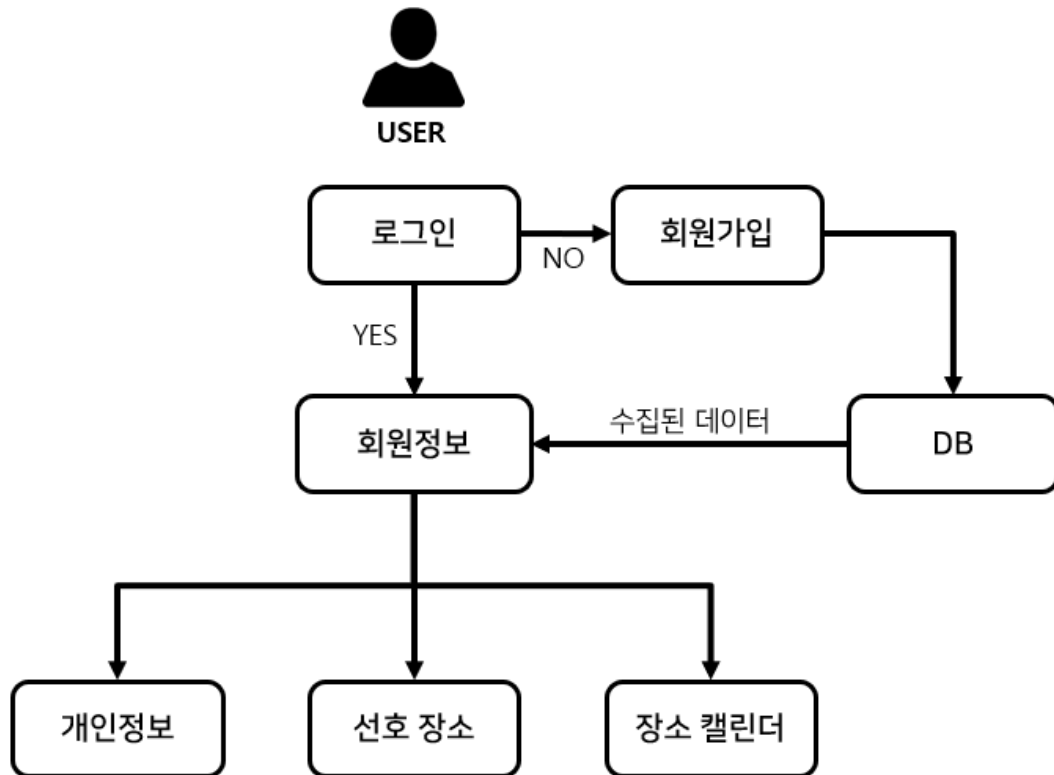
시스템 기능 모듈 상세설계(시각화)



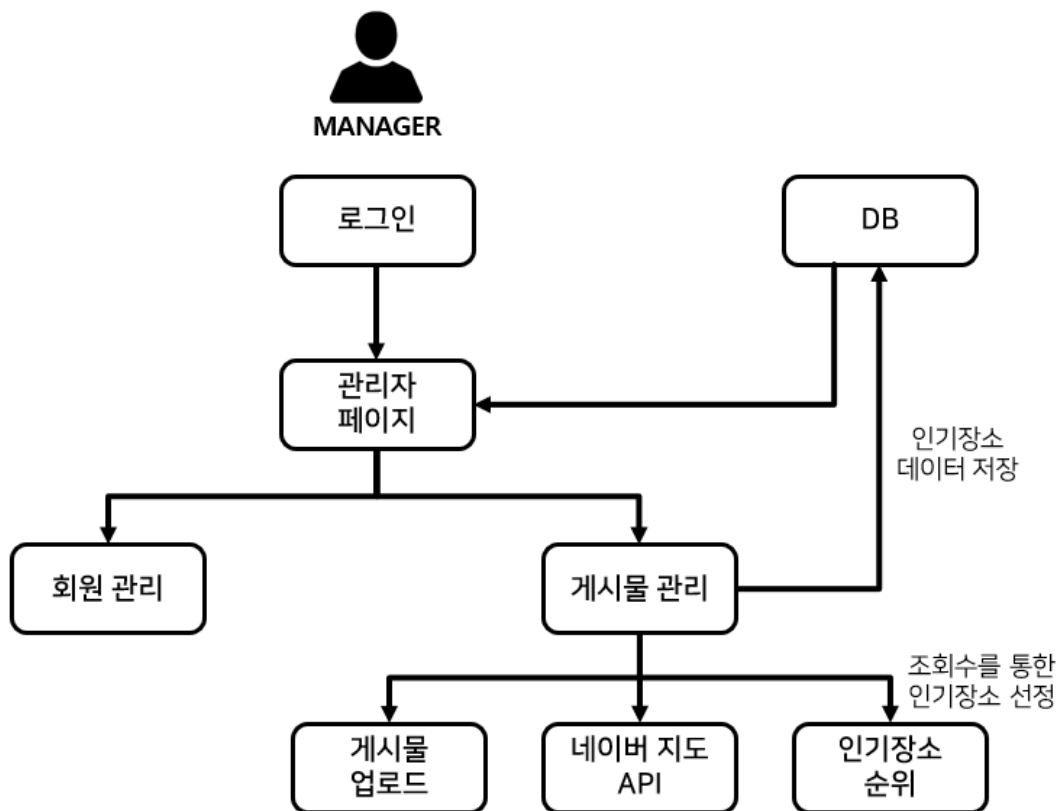
시스템 기능 모듈 상세설계(지도)



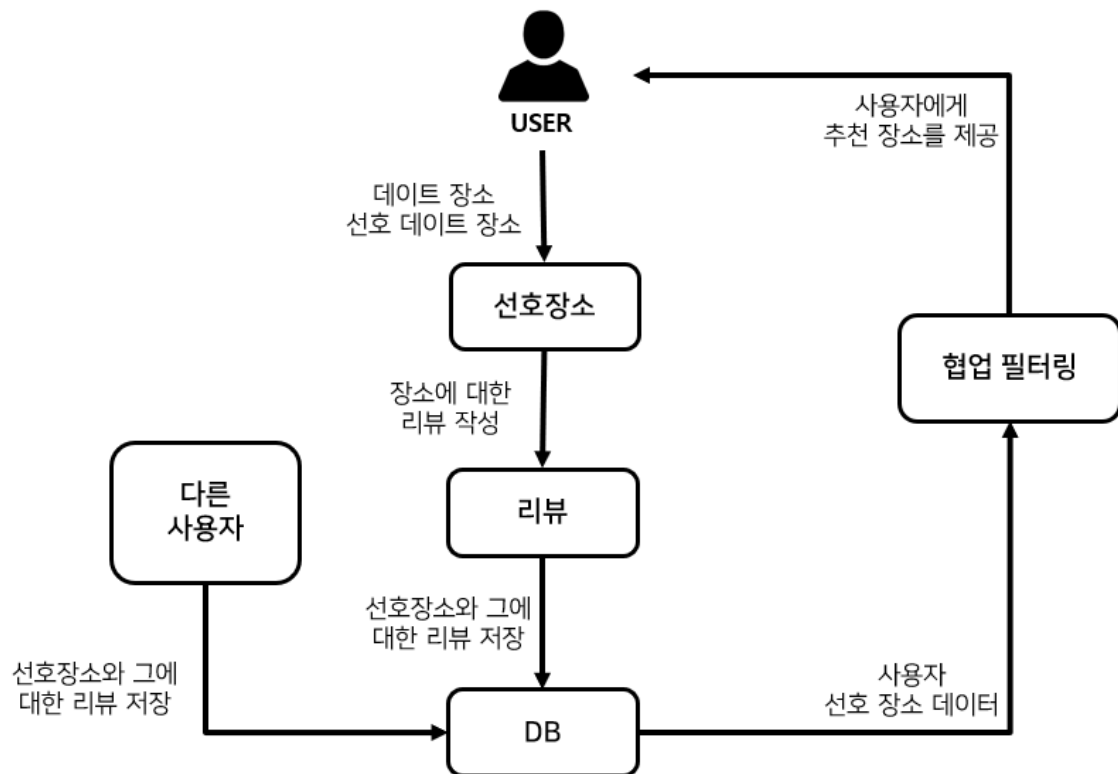
시스템 기능 모듈 상세설계(회원)



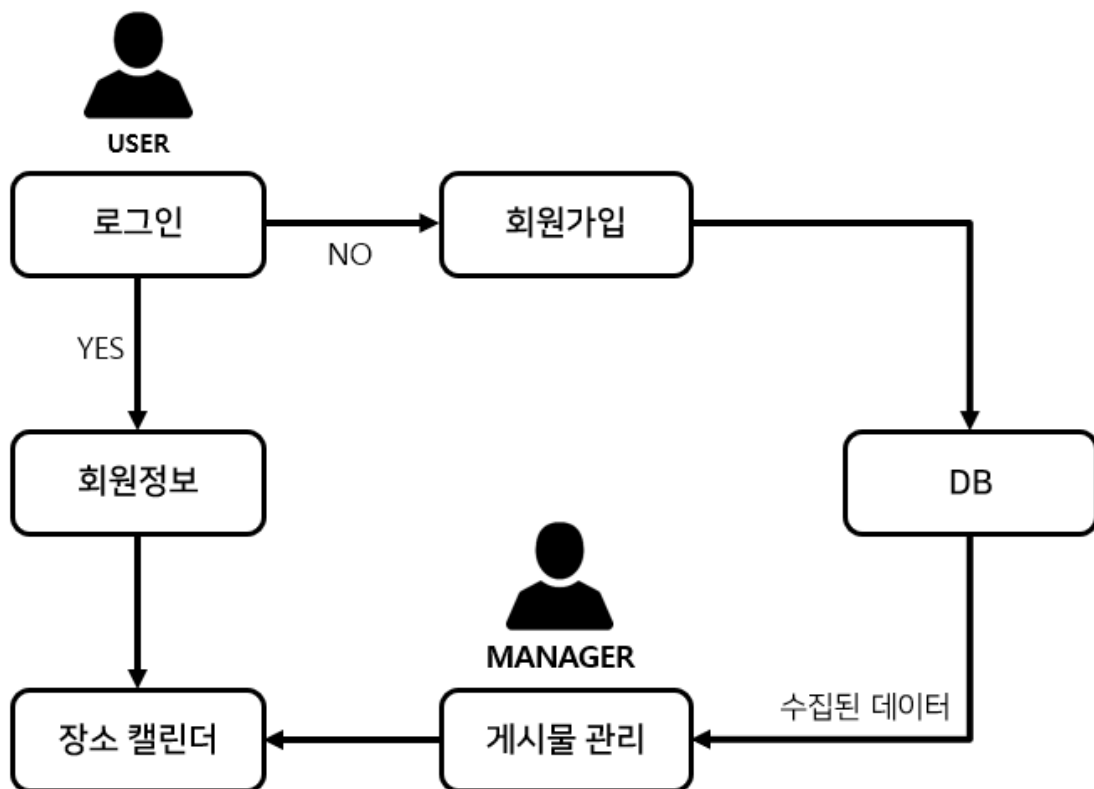
시스템 기능 모듈 상세설계(관리자)



시스템 기능 모듈 상세설계(추천장소)



시스템 기능 모듈 상세설계(장소캘린더)



함수명	내 용
introduce.html	<ul style="list-style-type: none"> • 사이트 로고, 관리자, 사이트를 소개하는 페이지
date_recommendation.html	<ul style="list-style-type: none"> • 주제별로 나눈 데이트 장소 추천하는 페이지
question.html	<ul style="list-style-type: none"> • 유저에게 문의를 받는 페이지 (회사주소, E-mail, 전화번호 등을 통해)
recommendation1.html	<ul style="list-style-type: none"> • 인기 순위 추천장소 • 1위부터 10위까지 인기 데이트 장소를 나열한 페이지
recommendation2.html	<ul style="list-style-type: none"> • 유저 추천장소 • 사이트 이용 유저의 장소 리뷰를 통한 좋아요 순으로 데이트 장소를 나열한 페이지
recommendation3.html	<ul style="list-style-type: none"> • 에디터 추천장소 • 관리자가 추천하는 장소나 계절에 맞는 장소를 추천하는 페이지
recommendation4.html	<ul style="list-style-type: none"> • 회원 맞춤형 추천장소 • 비슷한 데이트 장소를 선호하는 유저들을 비교하여 각각에게 선호하는 장소를 추천하는 페이지
privacy.html	<ul style="list-style-type: none"> • 유저 개인정보 페이지 • ID, 이름, 주소, 전화번호 등
calendar.html	<ul style="list-style-type: none"> • 장소캘린더 • 달력 형식으로 날씨, 계절을 고려하여 데이트 장소를 추천
signin.html	<ul style="list-style-type: none"> • 회원가입 페이지 • ID, Password, 이름, 주소, 전화번호 작성
signup.html	<ul style="list-style-type: none"> • 로그인 페이지
signout.html	<ul style="list-style-type: none"> • 로그아웃 페이지
manager.html	<ul style="list-style-type: none"> • 관리자 페이지 • 회원관리 페이지와 게시물 관리 페이지 접속
user_manager.html	<ul style="list-style-type: none"> • 회원관리 페이지 • DB와 연동하여 회원 데이터를 관리
post_manager.html	<ul style="list-style-type: none"> • 게시물 관리 페이지 • DB와 연동하여 게시물 데이터를 관리

함수명	내 용
crawler_last.py	<ul style="list-style-type: none"> • 크롤링 파이썬 파일 • Selenium과 Chrome driver를 사용하여 태그를 크롤링 한 후 .csv 파일로 저장
keyword_last.py	<ul style="list-style-type: none"> • 키워드 추출 파일
recommend.py	<ul style="list-style-type: none"> • 협업필터링을 사용한 데이트 장소 추천 파일

함수명	내용
crawler_last.py	
<pre> from urllib.parse import quote_plus from bs4 import BeautifulSoup from selenium import webdriver baseUrl = 'https://www.instagram.com/explore/tags/' plusUrl = input('검색어를 입력하세요 : ') url = baseUrl + quote_plus(plusUrl) driver = webdriver.Chrome() driver.get(url) ... soup = BeautifulSoup(driver.page_source, "html.parser") count_tag = soup.find('span',{'class': 'g47SY'}) ... try : data = driver.find_element_by_css_selector('.C7I1f.X7jCj') tag_raw = data.text tags = re.findall('#[A-Za-z0-9가-힣]+', tag_raw) tag = ".join(tags).replace("#"," ") tag_data = tag.split() for tag_one in tag_data : instagram_tags.append(tag_one) </pre>	<ul style="list-style-type: none"> • 크롤링 하기 위해 BeautifulSoup 와 selenium 라이브러리를 import 시켜줌 • 기본 인스타그램 태그의 주소와 검색어를 입력하는 Url주소를 합쳐서 검색기능을 구현 • import한 크롬드라이버로 접속 • 태그 크롤링 # .C7I1f.X7jCj 는 태그부분 페이지소스

함수명	내용
keyword_last.py	
<pre> from konlpy.tag import Okt from collections import Counter def get_tags(text, ntags=50): okt = Okt() nouns = okt.nouns(text) count = Counter(nouns) noun_count_list = [] ... for tag in tags: noun = tag['tag'] count = tag['count'] open_output_file.write('{} {} \n'.format(noun, count)) ... </pre>	<ul style="list-style-type: none"> • 키워드를 추출하기 위해 konlpy 라이브러리 사용 • 데이터 파일로부터 데이터를 가져온 후 리스트에 저장 • 리스트에서 tag, count만 추출

함수명	내용
recommend.py	
<pre> def sqrt_sum(v): length = len(v) sums = 0 for i in v: sums += i**2 return sqrt(sums) def cosine_simil(v1,v2): length = len(v1) sums = 0 for i in range(length): sums += v1[i]*v2[i] denominator = sqrt_sum(v1) * sqrt_sum(v2) if denominator: return sums / denominator else: return 0 def item_based_cosine_instructor_cf(): _,(iv,instr2idx,_) = get_item_vectors(*get_matrix2item(get_data())) num_instrs = len(iv) i_simils=[[[] for _ in range(num_instrs)] for _ in range(num_instrs)] print(len(i_simils)) for k,v in iv.items(): num_v = len(v[0]) if num_v < 2: continue i_from = int(k.split(',')[0]) idx_from = instr2idx[i_from] i_to = int(k.split(',')[1]) idx_to = instr2idx[i_to] simil = cosine_simil(v[0],v[1]) i_simils[idx_from][idx_to] = [real_round(simil,3),num_v] return i_simils </pre>	<ul style="list-style-type: none"> • sqrt_sum, 유사도를 계산하기 위한 분모 함수 • cosine_simil, 유사도 계산 • vectors 부분에서 중복된 값을 방지하기 위함 similarity = cosine / num_user

함수명	내용
Bootstrap	
<pre>STATIC_ROOT = os.path.join(BASE_DIR, 'static')</pre>	<ul style="list-style-type: none"> Django에서 js, css, img 등의 파일인 static 파일을 관리 해주는 static 폴더를 생성하고 프로젝트 폴더를 찾아갈 수 있게 settings.py에 추가 해주는 코드
<pre>{% load static %} <!DOCTYPE html> <html lang="en"> <head> ... <!-- Bootstrap core CSS --> <link href='{% static "blog/vendor/bootstrap/css/bootstrap.min.css" %}' rel="stylesheet"> <!-- Custom styles for this template --> <link href='{% static "blog/css/modern-business.css" %}' rel="stylesheet"> ... <!-- Bootstrap core JavaScript --> <script src='{% static "blog/vendor/jquery/jquery.min.js" %}'></script> <script src='{% static "blog/vendor/popper/popper.min.js" %}'></script> <script src='{% static "blog/vendor/bootstrap/js/bootstrap.min.js" %}'></script></pre>	<ul style="list-style-type: none"> static의 경로를 템플릿에서 인식할 수 있게하는 load static 코드와 Bootstrap CSS 파일과 bootstrap.js 파일을 사용하기 위해 필요한 link href 코드 설정
prjobject/urls.py	
<pre>from django.urls import path, include from django.contrib import admin urlpatterns = [path('admin/', admin.site.urls), path("", include('blog.urls')),]</pre>	<ul style="list-style-type: none"> blog.urls를 가져오는 행을 추가하는 것으로 들어오는 모든 접속 요청을 blog.url로 전송해 추가 명령을 찾게 해준다.
blog/urls.py	
<pre>from django.urls import path from . import views urlpatterns = [path("", views.index, name='index'), "" path("", views.recommendation4, name='recommendation'),]</pre>	<ul style="list-style-type: none"> view들의 루트 URL을 할당 시키는 것으로써 URL 패턴의 문자열에 매칭 시켜서 웹 사이트 접속 시에 주소로 접속 시 views.xxx로 가라고 인도 해준다. name은 url에 명칭을 부여한 것으로 view를 식별해주는 역할을 한다.
views.py	
<pre>def index(request) templates = loader.get_template('blog/index.html') context = { 'latest_question_list': "test", } return HttpResponse(template.render(context,request))</pre>	<ul style="list-style-type: none"> index 함수를 생성 하는 것으로써, request시에 이것을 넘겨받아 render method를 호출하고 호출받은 return을 되돌려 주어서 index(메인 사이트)를 보여주는 역할을 한다.

5. 프로토타입 설계

메인화면

Start Bootstrap

[Introduce](#) [Date](#) [FAQ](#) [Recommendation](#) [User](#)



에디터 선정 리스트



장소

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quos quisquam, error quod sed cumque, odio distinctio velit nostrum temporibus necessitatibus et facere atque iure perspiciatis mollitia recusandae vero vel quam!



장소

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam viverra euismod odio, gravida pellentesque urna varius vitae.



장소

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quos quisquam, error quod sed cumque, odio distinctio velit nostrum temporibus necessitatibus et facere atque iure perspiciatis mollitia recusandae vero vel quam!



장소

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam viverra euismod odio, gravida pellentesque urna varius vitae.



장소

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam viverra euismod odio, gravida pellentesque urna varius vitae.



장소

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Itaque earum nostrum suscipit ducimus nihil provident, perferendis rem illo, voluptate atque, sit eius in voluptates, nemo repellat fugiat excepturi! Nemo, esse.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Molestias, expedita, saepe, vero rerum deleniti beatae veniam harum neque nemo praesentium cum alias asperiores commodi.

Call to Action

에디터 추천 장소

Chicken POP

[Introduce](#) [Date](#) [FAQ](#) [Recommendation](#) [User](#)

에디터 데이트 장소 선정

[Home](#) / [Services](#)



태그1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quos quisquam, error quod sed cumque, odio distinctio velit nostrum temporibus necessitatibus et facere atque lure perspicatis mollitia recusandae vero vel quam!

[Learn More](#)



태그2

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam viverra euismod odio, gravida pellentesque urna varius vitae.

[Learn More](#)



태그3

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quos quisquam, error quod sed cumque, odio distinctio velit nostrum temporibus necessitatibus et facere atque lure perspicatis mollitia recusandae vero vel quam!

[Learn More](#)



태그4

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam viverra euismod odio, gravida pellentesque urna varius vitae.

[Learn More](#)



태그5

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam viverra euismod odio, gravida pellentesque urna varius vitae.

[Learn More](#)



태그6

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Itaque earum nostrum suscipit ducimus nihil provident, perferendis rem illo, voluptate atque, sit eius in voluptates, nemo repellat fugiat excepturi! Nemo, esse.

[Learn More](#)

Copyright © Your Website 2019

태그1 자세히 보기

Chicken POP

[Introduce](#) [Date](#) [FAQ](#) [Recommendation](#) [User](#)

인기 장소

[Home](#) / [Services](#) / [장소](#)



장소

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Reiciendis aliquid atque, nulla? Quos cum ex quis soluta, a laboriosam. Dicta expedita corporis animi vero voluptate voluptatibus possimus, veniam magni quis!

[Read More →](#)

Posted on January 1, 2020 by [Chicken POP](#)

[← Older](#) [Newer →](#)

Copyright © Your Website 2019

인기 순위 추천장소

[Home](#) / 인기 순위



1. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Laudantium veniam exercitationem expedita laborum at voluptate. Labore, voluptates totam at aut nemo deserunt rem magni pariatur quos perspiciatis atque eveniet unde.

[Learn More](#)



2. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Ut, odit velit cumque vero doloreque repellendus distinctio maiores rem expedita a nam vitae modi quidem similique ducimus! Velit, esse totam tempore.

[Learn More](#)



3. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Omnis, temporibus, dolores, at, praesentium ut unde repudiandae voluptatum sit ab debitis suscipit fugiat natus velit excepturi amet commodi deleniti alias possimus!

[Learn More](#)



4. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo, quidem, consectetur, officia rem officiis illum aliquam perspiciatis aspernatur quod modi hic nemo qui soluta aut eius fugit quam in suscipit?

[Learn More](#)



5. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Laudantium veniam exercitationem expedita laborum at voluptate. Labore, voluptates totam at aut nemo deserunt rem magni pariatur quos perspiciatis atque eveniet unde.

[Learn More](#)



6. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Laudantium veniam exercitationem expedita laborum at voluptate. Labore, voluptates totam at aut nemo deserunt rem magni pariatur quos perspiciatis atque eveniet unde.

[Learn More](#)



7. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Laudantium veniam exercitationem expedita laborum at voluptate. Labore, voluptates totam at aut nemo deserunt rem magni pariatur quos perspiciatis atque eveniet unde.

[Learn More](#)



8. 장소

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Laudantium veniam exercitationem expedita laborum at voluptate. Labore, voluptates totam at aut nemo deserunt rem magni pariatur quos perspiciatis atque eveniet unde.

[Learn More](#)

About 사이트 소개

Chicken POP

[introduce](#) [Services](#) [Contact](#) [Portfolio](#) [Blog](#) [Other Pages](#)

About Subheading

[Home](#) / [Introduce](#)



About Chicken POP

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed voluptate nihil eum consectetur similique? Consectetur, quod, incididunt, harum nisi dolores delectus reprehenderit voluptatem perferendis dicta dolorem non blanditis ex fugiat.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Saepe, magni, aperiam vitae illum voluptatum aut sequi impedit non velit ab ea pariatur sint quidem corporis eveniet. Odit, temporibus reprehenderit dolorum!

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Et, consequuntur, modi mollitia corporis ipsa voluptate corrupti eum ratione ex ea praesentium quibusdam? Aut, in eum facere corrupti necessitatibus perspiciatis quis?

Our Team



김효연

조장

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Possimus aut mollitia eum ipsum fugiat odio officis odit.

name@example.com



김기원

조원

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Possimus aut mollitia eum ipsum fugiat odio officis odit.

name@example.com

Copyright © Your Website 2019

FAQ

Chicken POP

[Introduce](#) [Date](#) [FAQ](#) [Recommendation](#) [User](#)

Company Contact

[Home](#) / [Contact](#)



상세 설명

Jeongwang-dong, Siheung-si, Gyeonggi-do,
15073, REP. OF KOREA

P: (123) 456-7890

E: name@example.com

H: Monday - Friday: 9:00 AM to 5:00 PM

Copyright © Your Website 2019

6. 시험 / 테스트 결과

파일명	결과값
-----	-----

파일명	결과값
-----	-----

crawler_last.py

```
C:\Users\Giwon\anaconda3\envs\project_crawling\python.exe
```

태그입력 : **데이트추천장소**

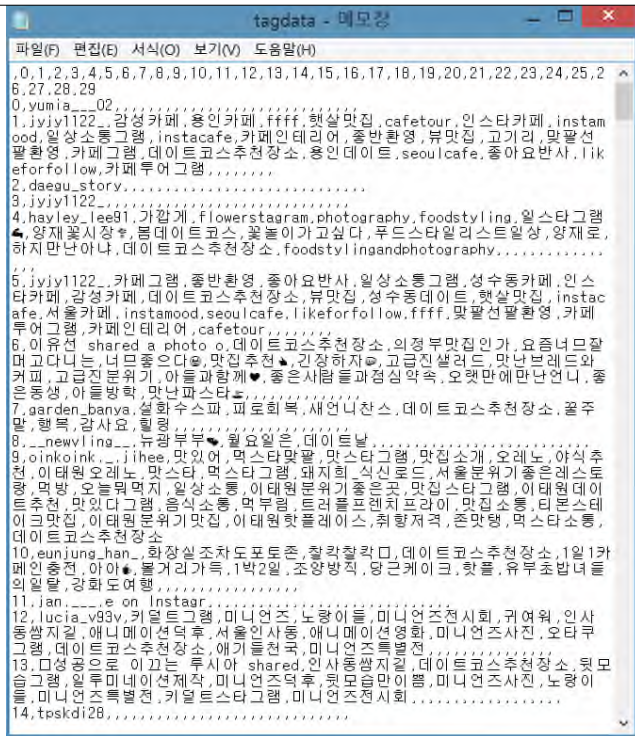
Chrome Driver 실행중...

총 4,102개의 게시물이 검색되었습니다.

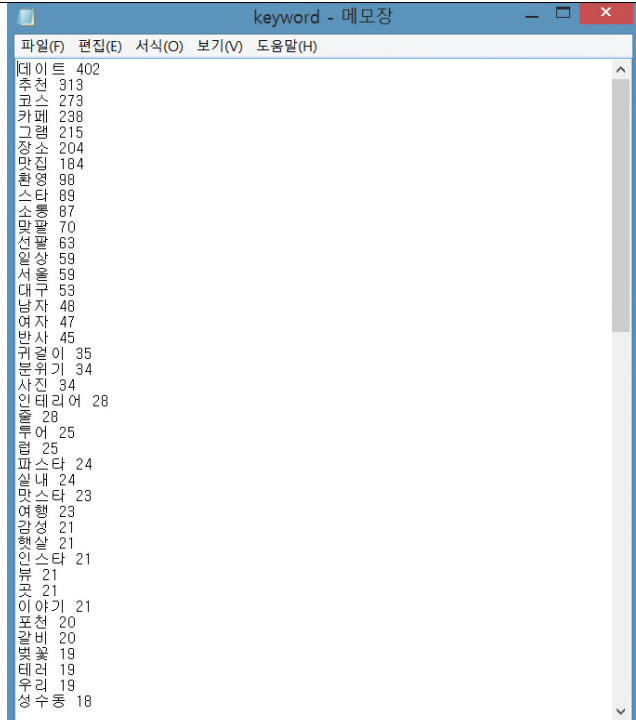
태그를 수집하는 중입니다...

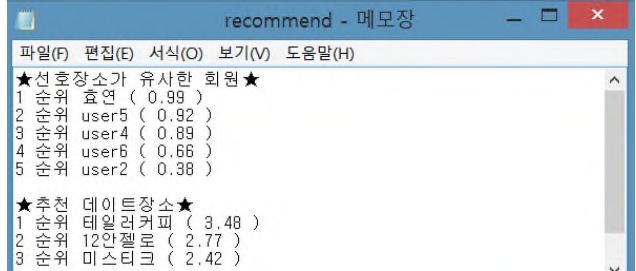
2208개의 태그를 수집합니다.

```
1%|          | 24/2208 [00:30<43:57, 1.21s/it]
```



	A	B	C	D	E	F	G	H	I	J	K	L	M
1			1	2	3	4	5	6	7	8	9	10	11
2		0 yumio_02											
3		1.jyy1122_ 강성제카	윤원익카	ffff		feat살맛	cafeleur	인스타트카	instamod	알리스트2	instacafe	카케인드	종반한원
4		2.daequ_story											
5		3.jyy1122_											
6		4.haylee_lee_가베카	flowerstg	photogr	foodstylin	일스타그	영양제카	꽃시본	데이트로	꽃들아	가도	부스터다	영양제
7		5.jyy1122_ 카케인드	종반한원	종아연반	알리스트2	성수통카	카케인드	강성제카	데이트로	부섯맛	성수통데	feat살맛	카케인드
8		6.yeyou_신데라이트로	의정양반	요요리온	디데리온	오맛지음	간헐적	가도	고급전설	가도	고급전분부	아름과글	제출은리온
9		7.arden_바삼하수피로	피복비	세아니찬	니데리온	고급지음	간헐적	간사요	형질				
10		8...newling_뉴강부마	데리온	데리온									
11		9.konkook... 카	엑스다	말타그	엑스다그린	엑스다	엑스다	엑스다	엑스다	엑스다	엑스다	엑스다	엑스다
12		10.enjung_노정함	노정함	노정함	노정함	노정함	노정함	노정함	노정함	노정함	노정함	노정함	노정함
13		12.hacia_y99카	기영	기영	기영	기영	기영	기영	기영	기영	기영	기영	기영
14		13.상승으로	인사동생	제이데이트	스모그	고급전분부	미네이	니언트	스모그	고급전분부	미네이	니언트	스모그
15		14.tpkd82n											
16		15.zx2020zzzz											
17		16.wjddk1k1	식후엔	디저트	맛도	수미카	카케인드	카케인드	카케인드	카케인드	카케인드	카케인드	카케인드
18		17.tuttu..._2											
19		18.tuttu..._1	맛있	제이	조희	예은	식음	식음	식음	식음	식음	식음	식음

파일명	결과값
keyword_last.py	 <pre> C:\Users\Giwon\anaconda3\envs\project_crawling\python.exe 키워드 추출 완료 Process finished with exit code 0 </pre>

파일명	결과값
recommend.py	 <pre> C:\Users\Giwon\anaconda3\envs\project_crawling\python.exe 추천 데이트 장소 추출 완료 Process finished with exit code 0 </pre>

메인화면

Chicken POP

Introduce Tag FAQ Recommendation User



전통 시장 탐방
커피 이색 데이트 전통 시장 탐방

에디터 선정 리스트



라플리에 김가
Unknown



언올드침
Unknown



카페인 신현리
Unknown



12안젤라
Unknown



미스티크
Unknown



테일러커피 연남1호점
Unknown

지역별 인기 장소



라세느

롯데호텔빌드의 세련된 뷔페 레스토랑 '라세느'에서는 계절별로 다채로운 요리를 즐길 수 있습니다. 한, 중, 일, 양식 등을 테마로 한 9개



에버랜드

1976년 개장한 에버랜드 리조트는 '플라워 카니발', '뽀머 스콜레쉬' 등 차별화된 축제와 엔터테인먼트로 국내 테마 파크의 수준을 선도하고

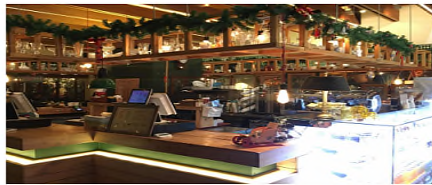


오이도 빨간등대

시흥시 오이도의 명물 빨간등대는 오이도가 2005년 정부의 '어촌체험 관광마을 조성사업' 대상 지역으로 선정되면서 거론됐다. 2006년

에디터 추천

[Home](#) / 에디터 추천



1. 빵공장 라플리에 김가 3.9

강원도 원주시 영구동 동구로 314
강원도 원주에 위치한 라플리에 김가는 비교적 사람이 없는 한적한 공간에 위치해 있으며 아늑하게 꾸며진 정원이 카페가 어우러져 사진이 잘 찍히는 명소로 유명하다.

[더 보기](#)



2. 언올드침 4.2

경기도 수원시 평달구 고학로 30번길 76
Unknown.

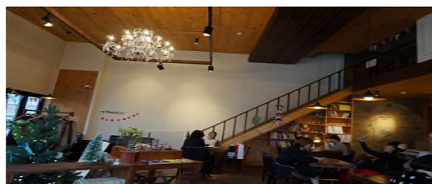
[더 보기](#)



3. 카페인 신현리 4.0

경기도 경주시 오포읍 새말길 167번길 68
Unknown

[더 보기](#)



4. 12안젤로 4.3

강원도 원주시 단계동 1115-3
Unknown

[더 보기](#)



5. 미스티크 4.1

경상남도 통영시 산양읍 면화리 194
Unknown

[더 보기](#)



6. 테일러커피 연남1호점 4.2

서울특별시 마포구 연남동 224-41
Unknown

[더 보기](#)



7. 장소

Unknown

[더 보기](#)



8. 장소

Unknown

[더 보기](#)

데이트장소 정보

Chicken POP

Introduce Tag FAQ Recommendation User

Home / [메디터 추천](#) / 라들리에 김가



라들리에 김가 3.9

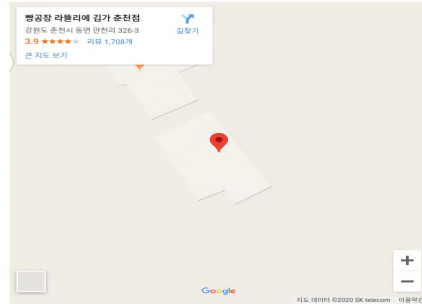
원주점

주소	강원도 원주시 행구동 행구로 314
전화번호	033-735-5677
종류	베이커리, 카페
주차	주차장(무보주차)
영업시간	10:00 ~ 22:00
쉬는시간	없음
메뉴 및 가격	점사의 정돈 7000 바게트 6000 트리플치즈 식빵 6000 치즈롤류 7000 레몬에이드 5000 아이스카라멜마끼아또 4500

Unknown

Posted on May 9, 2020 by [Chicken POP](#)

← Older Newer →



Copyright © Your Website 2019

사진 확대기능



장소캘린더

Calender

[⇐ Previous](#)
[⌵ Today](#)
[Next ⇒](#)

May 2017

Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
			<div>Event 1</div> <div>Event bla on same day!</div>		<div>Event 2</div>	
22	23	24	25	26	27	28
	<div>Event 3</div>					
29	30	31	<div>1</div> <div>Event 4</div>	2	3	4

7. Coding & Demo

■ 데이트 장소 데이터 수집

crawler_last.py

```
from urllib.request import urlopen, Request
from urllib.parse import quote_plus
from bs4 import BeautifulSoup
from selenium import webdriver
from tqdm import tqdm
import pandas as pd
import time
from selenium.webdriver.common.keys import Keys
import warnings
warnings.filterwarnings(action='ignore')

SCROLL_DOWN_TIME = 1.5 # 스크롤 내리는 시간
#=====
# Chrome Driver로 url 생성
baseUrl = "https://www.instagram.com/explore/tags/"
plusUrl = input('태그입력 : ')
url = baseUrl + quote_plus(plusUrl)

print('Chrome Driver 실행중...')
driver = webdriver.Chrome('C:\webdriver\chromedriver.exe')
driver.get(url)
time.sleep(3)

#=====
# 자동로그인
login_section = '//*[@id="react-root"]/section/nav/div[2]/div/div/div[3]/div/span/a[1]/button'
driver.find_element_by_xpath(login_section).click()
time.sleep(2)

elem_login = driver.find_element_by_name("username")
elem_login.clear()
elem_login.send_keys('ID') # 자신의 인스타그램 아이디

elem_login = driver.find_element_by_name('password')
elem_login.clear()
elem_login.send_keys('PASSWORD') # 자신의 인스타그램 비밀번호
time.sleep(2)

xpath = """/*[@id="react-root"]/section/main/div/article/div/div[1]/div/form/div[4]/button""
driver.find_element_by_xpath(xpath).click()
time.sleep(4)

#=====
# 총 게시물 숫자 불러오기
pageString = driver.page_source
bsObj = BeautifulSoup(pageString, 'lxml')
temp_data = bsObj.find_all(name='meta')[-1]
temp_data = str(temp_data)
```

```

start = temp_data.find('게시물') + 4
end = temp_data.find('개')
total_post_data = temp_data[start:end]
print("총 {0}개의 게시물이 검색되었습니다.".format(total_post_data))
print('태그를 수집하는 중입니다...')

#=====
reallink = [] # 전체 게시물 링크 저장

# url 화면에서 스크롤을 내리면서 게시물 크롤링
while True:
    pageString = driver.page_source
    bsObj = BeautifulSoup(pageString, 'lxml')

    for link1 in bsObj.find_all(name='div', attrs={"class": "Nnq7C weEfm"}):
        title = link1.select('a')[0]
        real = title.attrs['href']
        reallink.append(real)
        title = link1.select('a')[1]
        real = title.attrs['href']
        reallink.append(real)
        title = link1.select('a')[2]
        real = title.attrs['href']
        reallink.append(real)

    last_scroll = driver.execute_script('return document.body.scrollHeight')
    driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
    time.sleep(SCROLL_DOWN_TIME)
    new_scroll = driver.execute_script("return document.body.scrollHeight")

    if new_scroll == last_scroll:
        driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
        time.sleep(SCROLL_DOWN_TIME)
        new_scroll = driver.execute_script("return document.body.scrollHeight")

        # 현재 게시물이 마지막 게시물일 때 break
        if new_scroll == last_scroll:
            break
        else:
            last_scroll = new_scroll
            continue

    time.sleep(2)

total_tag_data = len(reallink) # 전체 태그 데이터의 크기 저장

print('{0}개의 태그를 수집합니다.'.format(total_tag_data))
csvtext = []

#=====
# 프로그램 실시간 진행률을 알기 위한 tqdm 라이브러리 사용
for i in tqdm(range(total_tag_data)):
    csvtext.append([])

```

```

req = Request("https://www.instagram.com/p"+reallink[i], headers={'User-Agent': 'Mozilla/5.0'})

webpage = urlopen(req).read()
soup = BeautifulSoup(webpage, 'lxml', from_encoding='utf-8')
soup1 = soup.find('meta', attrs={'property':"og:description"})

reallink1 = soup1['content']
reallink1 = reallink1[reallink1.find("@") + 1:reallink1.find("")]
reallink1 = reallink1[:20]

if reallink1 == "":
    reallink1 = "Null"
csvtext[i].append(reallink1)

for reallink2 in soup.find_all('meta', attrs={'property':"instapp:hashtags"}):
    hashtags = reallink2['content'].rstrip(',')
    csvtext[i].append(hashtags)

time.sleep(0.5)

# 데이터를 csv파일, txt파일로 저장
data = pd.DataFrame(csvtext)
data.to_csv('tagdata.csv', encoding='utf-8-sig')
data.to_csv('tagdata.txt', encoding='utf-8')

```

#

driver.close() # Chrome Driver 종료

keyword_last.py

```

#-*- coding:utf-8 -*-
from konlpy.tag import Okt
from collections import Counter

def get_tags(text, ntags=50):
    okt = Okt()
    nouns = okt.nouns(text)
    count = Counter(nouns)
    noun_count_list = [] # 빈도수를 저장하는 변수

    for n, c in count.most_common(ntags):
        temp = {'tag': n, 'count': c}
        noun_count_list.append(temp)

    return noun_count_list

input_file_name = "tagdata.txt"
noun_count = 100 # 빈도수에 따른 키워드 결과값 최대 100개
output_file_name = "keyword.txt"
open_text_file = open(input_file_name, 'r', -1, "utf-8")
text = open_text_file.read()
tags = get_tags(text, noun_count)
open_text_file.close()

```

```
open_output_file = open(output_file_name, 'w', -1, "utf-8")
```

```
for tag in tags:  
    noun = tag['tag']  
    count = tag['count']  
    open_output_file.write('{} {} \n'.format(noun, count))
```

```
open_output_file.close()
```

```
print("키워드 추출 완료")
```

```

#-*- coding:utf-8 -*-
import grade_data as d
import sys

from math import sqrt
#=====
# person1과 person2의 그래디언트 거리 계산
def sim_distance(prefs, person1, person2):
    si = dict()

    for item in prefs[person1]:
        if item in prefs[person2]:
            si[item] = 1

    if len(si) == 0:
        return 0

    sum_of_squares = sum([(prefs[person1][item] - prefs[person2][item])**2 for item in prefs[person1] if item in
prefs[person2]])

    return 1/(1+sqrt(sum_of_squares))

#=====

def sim_pearson(prefs, p1, p2):
    si = dict() # 같이 평가한 항목들의 목록을 구함

    for item in prefs[p1]:
        if item in prefs[p2]:
            si[item] = 1

    n = len(si) # 공통 항목 갯수

    # 공통 항목이 없으면 0
    if n == 0:
        return 0

    # 모든 선호도 합 계산
    sum1 = sum([prefs[p1][it] for it in si])
    sum2 = sum([prefs[p2][it] for it in si])

    # 제곱의 합 계산
    sum1Sq = sum([(prefs[p1][it])**2 for it in si])
    sum2Sq = sum([(prefs[p2][it])**2 for it in si])

    # 곱의 합 계산
    pSum = sum([prefs[p1][it] * prefs[p2][it] for it in si])

    # 피어슨 점수 계산
    num = pSum - (sum1*sum2/n)
    den = sqrt((sum1Sq-pow(sum1,2)/n) * (sum2Sq-pow(sum2,2)/n))
    if den==0:

```

```

    return 0

    r = num/den

    return r

#=====
# 선호도 dict를 사용하여 평가 유사도가 높은 상대편을 구함
def top_match(prefs, person, n=5, similarity=sim_pearson):
    scores = [ (similarity(prefs, person, other), other) for other in prefs if other!=person ]

    scores.sort()
    scores.reverse()
    return scores[:n]

#=====
# 다른 사람과의 순위의 가중 평균값을 이용해서 특정 사람을 추천
def get_recommendations(prefs, person, similarity=sim_pearson):
    total = dict()
    sim_Sum = dict()

    for other in prefs:
        if other == person:
            continue
        sim = similarity(prefs, person, other) #person과 other 사이의 유사도 점수를 구함

        # 0 이하 점수는 무시
        if sim <= 0:
            continue

        # other가 평가한 데이트 장소
        for item in prefs[other]:
            # 내가 평가하지 않은 장소만을 대상으로 함
            if item not in prefs[person] or prefs[person][item] == 0:
                # 유사도*점수
                total.setdefault(item, 0)
                total[item] += prefs[other][item]*sim # other가 평가한 영화의 점수 * person과 other의 상관계수
                # 유사도 합계
                sim_Sum.setdefault(item, 0)
                sim_Sum[item] += sim

    # 정규화된 목록 생성
    ranking = [ (total/sim_Sum[item], item) for item, total in total.items() ]

    # 정렬된 목록 리턴
    ranking.sort()
    ranking.reverse()

    return ranking

#=====

```

```
# 결과값 출력

print("추천 데이트 장소 추출 완료")

sys.stdout=open('recommend.txt', 'a') # 결과값을 텍스트파일로 저장

place_grade_rank=0
print("★ 선호장소가 유사한 회원★")

for score, other in top_match(d.grade, "기원"):
    place_grade_rank = place_grade_rank + 1
    print(place_grade_rank, "순위", other, "(", round(score, 2), ")")

print()

recommend_grade_rank = 0
print("★ 추천 데이트장소★")

for score, item in get_recommendations(d.grade, "기원", similarity=sim_distance):
    recommend_grade_rank = recommend_grade_rank + 1
    print(recommend_grade_rank, "순위", item, "(", round(score,2), ")")
```

views.py

```
from django.shortcuts import render, redirect, get_object_or_404
```

```
# Create your views here.
```

```
from django.http import HttpResponseRedirect, JsonResponse
from datetime import datetime
from django.template import loader
```

```
## Main page views
```

```
def index(request):
    template = loader.get_template('blog/index.html')
    now = datetime.now()
    context = {
        'latest_question_list': "test",
        'current_date': now,
    }
    return HttpResponseRedirect(template.render(context, request))
```

```
def introduce(request):
    return render(request, 'blog/introduce.html')
```

```
def services(request):
    return render(request, 'blog/services.html')
```

```
def contact(request):
    return render(request, 'blog/faq.html')
```

```
def recommendation1(request):
    return render(request, 'blog/recommendation1.html')
```

```
def recommendation2(request):
    return render(request, 'blog/recommendation2.html')
```

```
def recommendation3(request):
    return render(request, 'blog/recommendation3.html')
```

```
def recommendation4(request):
    return render(request, 'blog/recommendation4.html')
```

```
## Main Page Editor Recommendation views
```

```
def index_editor1(request):
    return render(request, 'blog/index/index_editor1.html')
```

```
def index_editor2(request):
    return render(request, 'blog/index/index_editor2.html')
```

```
def index_editor3(request):
    return render(request, 'blog/index/index_editor3.html')
```

```
## Recommendation One Page Recommendation views
```

```

def recommend3_1(request):
    return render(request, 'blog/recommend3/recommend3_1.html')

def recommend3_2(request):
    return render(request, 'blog/recommend3/recommend3_2.html')

def recommend3_3(request):
    return render(request, 'blog/recommend3/recommend3_3.html')

def recommend3_4(request):
    return render(request, 'blog/recommend3/recommend3_4.html')

def recommend3_5(request):
    return render(request, 'blog/recommend3/recommend3_5.html')

def recommend3_6(request):
    return render(request, 'blog/recommend3/recommend3_6.html')

def recommendation1_2(request):
    return render(request, 'blog/recommendation1_2.html')

## Service Page list views
def services_list1(request):
    return render(request, 'blog/services_list1.html')

```

sample_models.py

```

from django.db import models

class blog(models.Model):
    site_name = models.CharField(max_length=30)
    url = models.URLField()
    contents = models.TextField(blank=True)
    created = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return "Site name : " + self.site_name + ". URL: " + self.url

    class Meta:
        ordering = ["-created"]

# Create your models here.

```



```
{% load static %}
<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Modern Business - Start Bootstrap Template</title>

    <!-- Bootstrap core CSS -->
    <link href='{% static "blog/vendor/bootstrap/css/bootstrap.min.css" %}' rel="stylesheet">

    <!-- Custom styles for this template -->
    <link href='{% static "blog/css/modern-business.css" %}' rel="stylesheet">

</head>

<body>

    <!-- Navigation -->
    <nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-dark fixed-top">
        <div class="container-fluid">
            <a class="navbar-brand" href="{% url 'index' %}">Chicken POP</a>
            <button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse"
data-target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarResponsive">
                <ul class="navbar-nav ml-auto">
                    <li class="nav-item">
                        <a class="nav-link" href="/introduce">Introduce</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="/services">Tag</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="/contact">FAQ</a>
                    </li>
                    <li class="nav-item dropdown">
                        <a class="nav-link dropdown-toggle" href="#" id="navbarDropdownPortfolio" data-toggle="dropdown"
aria-haspopup="true" aria-expanded="false">
                            Recommendation
                        </a>
                        <div class="dropdown-menu dropdown-menu-right" aria-labelledby="navbarDropdownPortfolio">
                            <a class="dropdown-item" href="/recommendation1">인기 순위</a>
                            <a class="dropdown-item" href="/recommendation2">유저 추천</a>
                            <a class="dropdown-item" href="/recommendation3">에디터 추천</a>
                        </div>
                    </li>
                </ul>
            </div>
        </div>
    </nav>

```

```

        <a class="dropdown-item" href="/recommendation4">회원 맞춤형</a>
    </div>
</li>
<li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" id="navbarDropdownBlog" data-toggle="dropdown"
    aria-haspopup="true" aria-expanded="false">
        User
    </a>
    <div class="dropdown-menu dropdown-menu-right" aria-labelledby="navbarDropdownBlog">
        <a class="dropdown-item" href="/full-width">Sign in</a>
        <a class="dropdown-item" href="/sidebar">Sign up</a>
        <a class="dropdown-item" href="/faq">Privacy</a>
        <a class="dropdown-item" href="/404">Calendar</a>
    </div>
</li>
</ul>
</div>
</div>
</nav>

<header>
<div id="carouselExampleIndicators" class="carousel slide" data-ride="carousel">
    <ol class="carousel-indicators">
        <li data-target="#carouselExampleIndicators" data-slide-to="0" class="active"></li>
        <li data-target="#carouselExampleIndicators" data-slide-to="1"></li>
        <li data-target="#carouselExampleIndicators" data-slide-to="2"></li>
    </ol>
    <div class="carousel-inner" role="listbox">
        <!-- Slide One - Set the background image for this slide in the line below -->
        <div class="carousel-item active" style="background-image: url('{% static "blog/img/01.jpg" %}')">

            <div class="carousel-caption d-none d-md-block">
                <h3>전통 시장 탐방</h3>
                <p>커플 이색 데이트 전통 시장 탐방.</p>
            </div>
        </div>
        <!-- Slide Two - Set the background image for this slide in the line below -->
        <div class="carousel-item" style="background-image: url('{% static "blog/img/02.jpg" %}')">
            <div class="carousel-caption d-none d-md-block">
                <h3>양식 인기 맛집</h3>
                <p>고품격 레스토랑 부터 캐주얼까지 총정리.</p>
            </div>
        </div>
        <!-- Slide Three - Set the background image for this slide in the line below -->
        <div class="carousel-item" style="background-image: url('{% static "blog/img/03.jpg" %}')">
            <div class="carousel-caption d-none d-md-block">
                <h3>연말 행사 장소 정리</h3>
                <p>파티나 모임 연회에 안성맞춤인 장소</p>
            </div>
        </div>
    </div>
    <a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-slide="prev">
        <span class="carousel-control-prev-icon" aria-hidden="true"></span>

```

```

        <span class="sr-only">Previous</span>
    </a>
    <a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-slide="next">
        <span class="carousel-control-next-icon" aria-hidden="true"></span>
        <span class="sr-only">Next</span>
    </a>
</div>
</header>

<!-- Page Content -->
<div class="container">

    <h1 class="my-4"></h1>

    <!-- Portfolio Ssection -->
    <h2>에디터 선정 리스트</h2>

    <div class="row">
        <div class="col-lg-4 col-sm-6 portfolio-item">
            <div class="card h-100">
                <a href="/recommend1_1"></a>
                <div class="card-body">
                    <h4 class="card-title">
                        <a href="/recommend3_1">라플리에 김가</a>
                    </h4>
                    <p class="card-text">Unknown</p>
                </div>
            </div>
        </div>
        <div class="col-lg-4 col-sm-6 portfolio-item">
            <div class="card h-100">
                <a href="/recommend3_2"></a>
                <div class="card-body">
                    <h4 class="card-title">
                        <a href="/recommend3_2">언올드침</a>
                    </h4>
                    <p class="card-text">Unknown</p>
                </div>
            </div>
        </div>
        <div class="col-lg-4 col-sm-6 portfolio-item">
            <div class="card h-100">
                <a href="/recommend3_3"></a>
                <div class="card-body">
                    <h4 class="card-title">
                        <a href="/recommend3_3">카페인 신현리</a>
                    </h4>
                    <p class="card-text">Unknown</p>
                </div>
            </div>
        </div>
    </div>

```

```

</div>
<div class="col-lg-4 col-sm-6 portfolio-item">
  <div class="card h-100">
    <a href="/recommend3_4"></a>
    <div class="card-body">
      <h4 class="card-title">
        <a href="/recommend3_4">12안젤라</a>
      </h4>
      <p class="card-text">Unknown</p>
    </div>
  </div>
</div>
</div>
<div class="col-lg-4 col-sm-6 portfolio-item">
  <div class="card h-100">
    <a href="/recommend3_5"></a>
    <div class="card-body">
      <h4 class="card-title">
        <a href="/recommend3_5">미스틱크</a>
      </h4>
      <p class="card-text">Unknown</p>
    </div>
  </div>
</div>
</div>
<div class="col-lg-4 col-sm-6 portfolio-item">
  <div class="card h-100">
    <a href="/recommend3_6"></a>
    <div class="card-body">
      <h4 class="card-title">
        <a href="/recommend3_6">테일러커피 연남1호점</a>
      </h4>
      <p class="card-text">Unknown</p>
    </div>
  </div>
</div>
</div>
</div>
<!-- /.row -->

```

<h2>지역별 인기 장소</h2>

```

<div class="row">
  <div class="col-lg-4 col-sm-6 portfolio-item">
    <div class="card h-100">
      <a href="/index_editor1"></a>
      <div class="card-body">
        <h4 class="card-title">
          <a href="/index_editor1">라세느</a>
        </h4>
        <p class="card-text">롯데호텔월드의 세련된 뷔페 레스토랑 ‘라세느’에서는 계절별로 다채로운 요리
를 즐길 수 있습니다. 한, 중, 일, 양식 등을 테마로 한 9개의 즉석요리 코너에서 150여 가지의 다양한 요리를 맛볼

```

수 있습니다. 또한, 계절별 음식재료를 사용한 신선한 요리와 매월 '이벤트 메뉴'로 제공되는 세계 여러 나라의 다채로운 음식을 경험할 수 있습니다.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-sm-6 portfolio-item">

<div class="card h-100">

<div class="card-body">

<h4 class="card-title">

에버랜드

</h4>

<p class="card-text">1976년 개장한 에버랜드 리조트는 '플라워 카니발', '썸머 스플래쉬' 등 차별화된 축제와 엔터테인먼트로 국내 테마 파크의 수준을 선도하고 있으며, 세계적 규모의 워터 파크, 스포츠 시설, 골프 클럽으로 손님 여러분께 늘 휴일 같은 편안함과 즐거움을 선사하고 있습니다.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-sm-6 portfolio-item">

<div class="card h-100">

<div class="card-body">

<h4 class="card-title">

오이도 빨간등대

</h4>

<p class="card-text">시흥시 오이도의 명물 빨간등대는 오이도가 2005년 정부의 '어촌체험 관광마을 조성사업' 대상 지역으로 선정되면서 건립됐다. 2006년 드라마 '여우야 뭐하니'에 등장하면서, 빨간등대는 연인들의 데이트코스, 사진작가들의 출사지에 꼭 포함되는 명소로 각광받기 시작했다.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-sm-6 portfolio-item">

<div class="card h-100">

<div class="card-body">

<h4 class="card-title">

롯데월드 아쿠아리움

</h4>

<p class="card-text">사람과 자연이 더불어 사는 즐거운 세상을 꿈꾸는 롯데월드 아쿠아리움. 깊고 드넓은 해양 생태계 그대로를 재현하겠다는 신념과 고집으로 지금까지의 아쿠아리움에 대한 모든 기준을 버리고 해양생물의 입장이 되어, 당신의 입장이 되어 원점에서부터 시작합니다.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-sm-6 portfolio-item">

<div class="card h-100">

<div class="card-body">

<h4 class="card-title">

바스버거

</h4>

<p class="card-text">2015년 여의도 본점을 시작해서 현재는 총 10개 매장을 운영하고 있는 바스버거는 줄서서 먹는 햄버거집으로 유명하며 나만 알고 싶었던 맛집에서 모두가 아는 맛집으로 변화해 현재 모두에게 사

```

    량받고 있다.</p>
    </div>
  </div>
</div>
<div class="col-lg-4 col-sm-6 portfolio-item">
  <div class="card h-100">
    <a href="#"></a>
    <div class="card-body">
      <h4 class="card-title">
        <a href="#">팩피</a>
      </h4>
      <p class="card-text">2년 연속으로 미술랭에 선정된 Freaking Awesome Good Pasta라는 뜻의 팩피. 파
스타를 전문으로 하는 곳임에도 전석이 바 테이블로 되어있는 점이 재미있고, 매장은 협소하지만 오픈 키친이 크게
구성된 점이 흥미로운 가게이다.</p>
    </div>
  </div>
</div>
</div>

<!-- Footer -->
<footer class="py-5 bg-dark">
  <div class="container">
    <p class="m-0 text-center text-white">Copyright &copy; Your Website 2019</p>
  </div>
<!-- /.container -->
</footer>

<!-- Bootstrap core JavaScript -->
<script src="{% static "blog/vendor/jquery/jquery.min.js" %}"></script>
<script src="{% static "blog/vendor/popper/popper.min.js" %}"></script>
<script src="{% static "blog/vendor/bootstrap/js/bootstrap.min.js" %}"></script>

</body>

</html>

```

```
#URLconf file
from django.urls import path
from django.conf.urls import url
from django.conf.urls.static import static
from django.conf import settings
from . import views

urlpatterns = [
    path('', views.index, name='index'),
    path('introduce/', views.introduce, name='introduce'),
    path('services/', views.services, name='services'),
    path('services_list1', views.services_list1, name='services_list1'),
    path('contact/', views.contact, name='contact'),
    path('recommendation1/', views.recommendation1, name='recommendation1'),
    path('recommendation1_2/', views.recommendation1_2, name='recommendation1_2'),
    path('recommendation2/', views.recommendation2, name='recommendation2'),
    path('recommendation3/', views.recommendation3, name='recommendation3'),
    path('recommendation4/', views.recommendation4, name='recommendation4'),
    path('index_editor1/', views.index_editor1, name='index_editor1'),
    path('index_editor2/', views.index_editor2, name='index_editor2'),
    path('index_editor3/', views.index_editor3, name='index_editor3'),
    path('recommend3_1/', views.recommend3_1, name='recommend3_1'),
    path('recommend3_2/', views.recommend3_2, name='recommend3_2'),
    path('recommend3_3/', views.recommend3_3, name='recommend3_3'),
    path('recommend3_4/', views.recommend3_4, name='recommend3_4'),
    path('recommend3_5/', views.recommend3_5, name='recommend3_5'),
    path('recommend3_6/', views.recommend3_6, name='recommend3_6'),
]
```

```
{% load static %}
<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Modern Business - Start Bootstrap Template</title>

    <!-- Bootstrap core CSS -->
    <link href='{% static "blog/vendor/bootstrap/css/bootstrap.min.css" %}' rel="stylesheet">

    <!-- Custom styles for this template -->
    <link href='{% static "blog/css/modern-business.css" %}' rel="stylesheet">

    <!-- Core CSS file -->
    <link href='{% static "blog/css/photoswipe.css"%}' rel="stylesheet">

    <!-- Skin CSS file (styling of UI - buttons, caption, etc.)
        In the folder of skin CSS file there are also:
        - .png and .svg icons sprite,
        - preloader.gif (for browsers that do not support CSS animations) -->
    <link href='{% static "blog/css/default-skin/default-skin.css"%}' rel="stylesheet" >

    <script src="https://code.jquery.com/jquery-3.2.1.min.js"></script>

</head>

<body>

    <!-- Navigation -->
    <nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-dark fixed-top">
        <div class="container-fluid">
            <a class="navbar-brand" href="{% url 'index' %}">Chicken POP</a>
            <button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse"
data-target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarResponsive">
                <ul class="navbar-nav ml-auto">
                    <li class="nav-item">
                        <a class="nav-link" href="/introduce">Introduce</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="/services">Tag</a>
                    </li>
                    <li class="nav-item">
```

```

    <a class="nav-link" href="/contact">FAQ</a>
  </li>
  <li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" id="navbarDropdownPortfolio" data-toggle="dropdown"
aria-haspopup="true" aria-expanded="false">
      Recommendation
    </a>
    <div class="dropdown-menu dropdown-menu-right" aria-labelledby="navbarDropdownPortfolio">
      <a class="dropdown-item" href="/recommendation1">인기 순위</a>
      <a class="dropdown-item" href="/recommendation2">유저 추천</a>
      <a class="dropdown-item" href="/recommendation3">에디터 추천</a>
      <a class="dropdown-item" href="/recommendation4">회원 맞춤형</a>
    </div>
  </li>
  <li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" id="navbarDropdownBlog" data-toggle="dropdown"
aria-haspopup="true" aria-expanded="false">
      User
    </a>
    <div class="dropdown-menu dropdown-menu-right" aria-labelledby="navbarDropdownBlog">
      <a class="dropdown-item" href="/full-width">Sign in</a>
      <a class="dropdown-item" href="/sidebar">Sign up</a>
      <a class="dropdown-item" href="/faq">Privacy</a>
      <a class="dropdown-item" href="/404">Calendar</a>
    </div>
  </li>
</ul>
</div>
</div>
</nav>

```

```

<!-- Page Content -->
<div class="container-fluid">

```

```

  <!-- Page Heading/Breadcrumbs -->
  <a><br></a>
  <ol class="breadcrumb">
    <li class="breadcrumb-item">
      <a href="{% url 'index' %}">Home</a>
    </li>
    <li class="breadcrumb-item">
      <a href="/recommendation3">에디터 추천</a>
    </li>
    <li class="breadcrumb-item active">라플리에 김가</li>
  </ol>

```

```

<div class="row">

```

```

  <!-- Blog Entries Column -->
  <div class="col-6">

```

```

    <!-- Blog Post -->
    <div class="card mb-4">

```

```


<div class="card-body">
  <h1>라뜰리에 김가<span style="font-weight: bold; font-size: x-large; color:
#FFA500;">&nbsp;3.9</span></h1>
  <p style="color: #AEAEAE;">&nbsp;원주점</p>
  <table class="table">
    <thead>
      <tr>
        <td style="color: #D5D4D4;" scope="col">주소</td>
        <td scope="col">강원도 원주시 행구동 행구로 314</td>
      </tr>
    </thead>
    <thead>
      <tr>
        <td style="color: #D5D4D4;" scope="col">전화번호</td>
        <td scope="col">033-735-5677</td>
      </tr>
    </thead>
    <thead>
      <tr>
        <td style="color: #D5D4D4;" scope="col">종류</td>
        <td scope="col">베이커리, 카페</td>
      </tr>
    </thead>
    <thead>
      <tr>
        <td style="color: #D5D4D4;" scope="col">주차</td>
        <td scope="col">주차장(무료주차)</td>
      </tr>
    </thead>
    <thead>
      <tr>
        <td style="color: #D5D4D4;" scope="col">영업시간</td>
        <td scope="col">10:00 ~ 22:00</td>
      </tr>
    </thead>
    <thead>
      <tr>
        <td style="color: #D5D4D4;" scope="col">쉬는시간</td>
        <td scope="col">없음</td>
      </tr>
    </thead>
    <thead>
      <tr>
        <td style="color: #D5D4D4;" scope="col">메뉴 및 가격</td>
        <td scope="col">천사의 질투 7000<br>바게트 6000<br>트리플치즈 식빵 6000<br>치즈퐁듀
7000<br>레몬에이드 5000<br>아이스카라멜마끼야또 4500</td>
      </tr>
    </thead>
  </table>
  <p class="card-text">Unknown</p>
</div>

```

```

<div class="card-footer text-muted">
  Posted on May 9, 2020 by
  <a href="#">Chicken POP</a>
</div>
</div>

<!-- Pagination -->
<ul class="pagination justify-content-center mb-4">
  <li class="page-item">
    <a class="page-link">&larr; Older</a>
  </li>
  <li class="page-item">
    <a class="page-link" href="/recommend3_2">Newer &rarr;</a>
  </li>
</ul>

</div>

<div class="col-4">
  <!-- Embedded Google Map -->
  <
    i
    f
    r
    a
    m
    e
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d393.62131600617084!2d127.7779145542556!3d37.88437192
2565016!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x3562e562c130f291%3A0xe3040baf709c551!2z67ml6rO17J6II
OudvOucsOumrOyXkCDquYDqsIAg7LaY7LKc7KCQ!5e0!3m2!1sko!2skr!4v1589097477320!5m2!1sko!2skr" width="600"
height="590" frameborder="0" style="border:0;" allowfullscreen="" aria-hidden="false" tabindex="0"></iframe>
  <a><br></a>
  <table style="border-spacing: 10px;">
    <tr>
      <td scope="col"></td>
      <td scope="col"></td>
    </tr>
    <tr>
      <td scope="col"></td>
      <td scope="col"></td>
    </tr>
  </table>

</div>

<!-- Root element of PhotoSwipe. Must have class pswp. -->
<div class="pswp" tabindex="-1" role="dialog" aria-hidden="true">

  <!-- Background of PhotoSwipe.
  It's a separate element as animating opacity is faster than rgba(). -->
  <div class="pswp__bg"></div>

  <!-- Slides wrapper with overflow:hidden. -->

```

```

<div class="pswp__scroll-wrap">

    <!-- Container that holds slides.
    PhotoSwipe keeps only 3 of them in the DOM to save memory.
    Don't modify these 3 pswp__item elements, data is added later on. -->
    <div class="pswp__container">
        <div class="pswp__item"></div>
        <div class="pswp__item"></div>
        <div class="pswp__item"></div>
    </div>

    <!-- Default (PhotoSwipeUI_Default) interface on top of sliding area. Can be changed. -->
    <div class="pswp__ui pswp__ui--hidden">

        <div class="pswp__top-bar">

            <!-- Controls are self-explanatory. Order can be changed. -->

            <div class="pswp__counter"></div>

            <button class="pswp__button pswp__button--close" title="Close (Esc)"></button>

            <button class="pswp__button pswp__button--share" title="Share"></button>

            <button class="pswp__button pswp__button--fs" title="Toggle fullscreen"></button>

            <button class="pswp__button pswp__button--zoom" title="Zoom in/out"></button>

            <!-- Preloader demo https://codepen.io/dimsemenov/pen/yyBWor -->
            <!-- element will get class pswp__preloader--active when preloader is running -->
            <div class="pswp__preloader">
                <div class="pswp__preloader__icn">
                    <div class="pswp__preloader__cut">
                        <div class="pswp__preloader__donut"></div>
                    </div>
                </div>
            </div>

            <div class="pswp__share-modal pswp__share-modal--hidden pswp__single-tap">
                <div class="pswp__share-tooltip"></div>
            </div>
            <button class="pswp__button pswp__button--arrow--left" title="Previous (arrow left)">
            </button>

            <button class="pswp__button pswp__button--arrow--right" title="Next (arrow right)">
            </button>

            <div class="pswp__caption">
                <div class="pswp__caption__center"></div>
            </div>

        </div>
    </div>

```

```
</div>

</div>

<script>
var pswpElement = document.querySelectorAll('.pswp')[0];

// build items array
var items = [
  {
    src: 'https://placekitten.com/600/400',
    w: 600,
    h: 400
  },
  {
    src: 'https://placekitten.com/1200/900',
    w: 1200,
    h: 900
  }
];

// define options (if needed)
var options = {
  // optionName: 'option value'
  // for example:
  index: 0 // start at first slide
};

// Initializes and opens PhotoSwipe
var gallery = new PhotoSwipe( pswpElement, PhotoSwipeUI_Default, items, options);
gallery.init();
</script>
<!-- /.container -->

<!-- Footer -->
<footer class="py-5 bg-dark">
  <div class="container">
    <p class="m-0 text-center text-white">Copyright &copy; Your Website 2019</p>
  </div>
<!-- /.container -->
</footer>

<!-- Bootstrap core JavaScript -->
<script src='{% static "blog/vendor/jquery/jquery.min.js" %}'></script>
<script src='{% static "blog/vendor/bootstrap/js/bootstrap.bundle.min.js" %}'></script>

<!-- Core JS file -->
<script src='{% static "blog/js/photoswipe.min.js"%}'></script>
<!-- UI JS file -->
<script src='{% static "blog/js/photoswipe-ui-default.min.js"%}'></script>
</body>

</html>
```

III. 결론

1. 연구결과

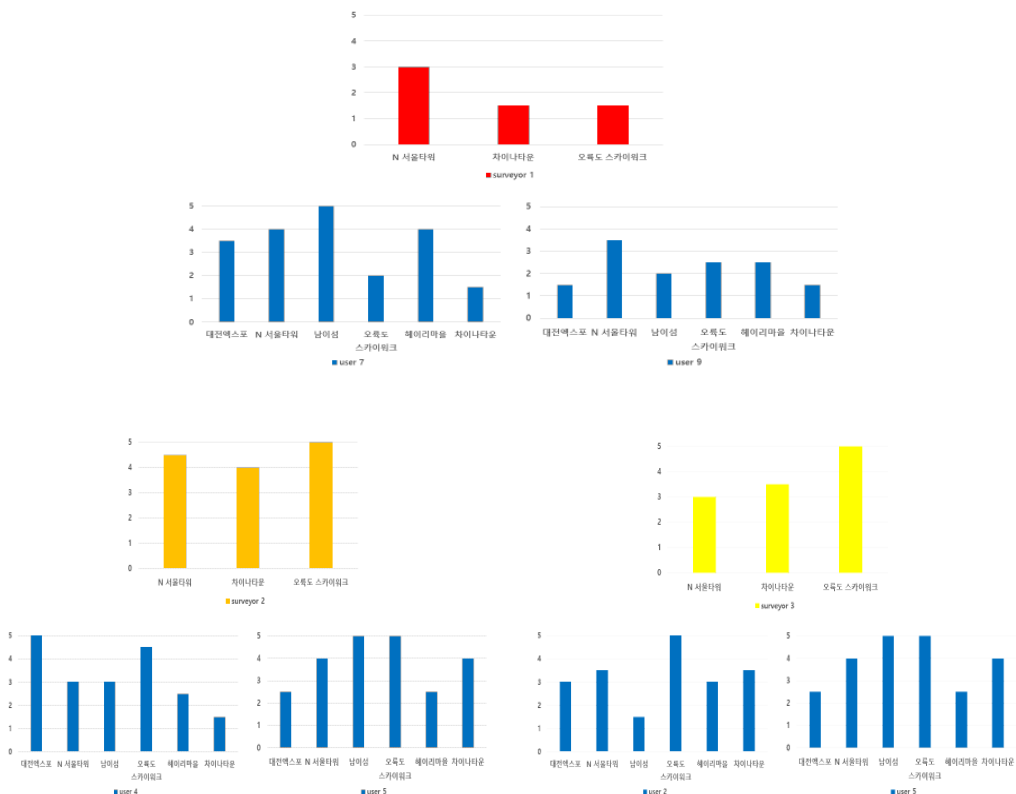
빅데이터 분석 결과

- 맞춤형 추천 데이트 장소를 제공하기 위하여 협업 필터링 알고리즘을 사용하여 빅데이터를 분석하였다. 총 20명을 대상으로 조사를 하였다. 그림과 같이 user 10명에게 데이트 장소에 대한 평점을 매겼고, Surveyor 10명의 회원들을 대상으로 데이트 장소, 3곳에 대한 평점을 수집했다.

```
'user1': {  
  '대전엑스포 시민광장': 2.5,  
  'N 서울타워': 3.5,  
  '남이섬': 3.0,  
  '오륙도 스카이워크': 3.5,  
  '차이나타운': 2.5,  
  '해이리마을': 3.0  
},  
'user2': {  
  '대전엑스포 시민광장': 3.0,  
  'N 서울타워': 3.5,  
  '남이섬': 1.5,  
  '오륙도 스카이워크': 5.0,  
  '해이리마을': 3.0,  
  '차이나타운': 3.5  
},  
'Surveyor1': {  
  'N 서울타워': 3.0,  
  '차이나타운': 1.5,  
  '오륙도 스카이워크': 1.5  
},  
'Surveyor2': {  
  'N 서울타워': 4.5,  
  '차이나타운': 4.0,  
  '오륙도 스카이워크': 5.0  
},
```

<수집된 평점 데이터>

- 하단의 그림과 같이 수집한 데이터를 막대그래프로 만들어서 비교하였다. 그 결과 선호 장소가 유사한 조사자들에게 협업필터링 알고리즘으로 도출된 데이터로 추천 데이트 장소를 제공했을 때 10명 중 7명의 조사자들이 긍정을 표해 약 70%의 일치율이 보이는 것을 확인했다.





<데이터 비교 그래프>

최종결론

- 데이터 크롤링을 이용하여 데이트 장소 데이터를 수집, 키워드분석 및 협업필터링 프로그램을 사용하여 데이터를 분석 및 정제를 설명 하였다. 이를 위하여 Python 라이브러리인 BeautifulSoup와 Selenium을 활용하여 크롤링 프로그램을 작성하였으며 빈도수 저장 및 그레디언트 거리 계산을 사용하여 키워드분석 및 협업필터링 프로그램을 구현하였다.
- 데이트 장소 데이터 접근에 관한 용이성을 제공하기 위해서 웹 사이트를 구현하였다. 태그별 기능을 통해 분류 정리된 데이터에 사용자가 관심을 가지는 태그에 쉽게 접근 할 수 있게 하였으며, 선호 장소 체크 기능을 활용하여 개개인의 사용자의 선호 장소를 한 눈에 확인할 수 있도록 구현 하였다.
- 본 시스템은 데이트 추천 장소 데이터를 집약 시켜서 웹 사이트로 구현한 시스템으로 기존에는 데이트 장소 추천 웹 사이트의 부재를 통해서 관련 정보를 획득하는 것에 대한 어려움을 겪었지만 본 시스템을 통해서 데이트 장소 선택에 관한 어려움을 겪는 사용자들에게 선택의 용이성을 기여할 수 있을 것이다.
- 향후 데이트 장소 수집 프로그램의 자동화 및 보다 나은 추천 시스템의 개발을 통해 본 시스템과 결합하여 서비스가 가능 하도록 발전이 가능할 것이라 본다.

2. 작품제작 소요재료 목록

소요재료	내 용
	<ul style="list-style-type: none">• 노동력
	<ul style="list-style-type: none">• 시간

참고자료

순번	참고문헌 정보
1	<ul style="list-style-type: none"> • 모바일 상황인식 추천맛집 서비스 개발 (2007 한국콘텐츠학회 논문지, 류종민, 홍창표, 강경보, 양두영, 좌정우)
2	<ul style="list-style-type: none"> • 빅데이터 기반 사용자 맞춤 헬스케어, 웰니스 추천 서비스 (2018.05 국가과학기술정보센터, 이종찬, 이문호)
3	<ul style="list-style-type: none"> • 상품 평가 분석 및 사용자 구매 기준을 반영한 상품 추천 시스템 (2019.05 배재대학교 대학원, 박재희)
4.	<ul style="list-style-type: none"> • Adaptive web search based on user profile constructed without any effort from users (2004.05 Nara Institute of Science and Technology, Kazunari Sugiyama 외 2명)
5	<ul style="list-style-type: none"> • Context-aware intelligent recommendation system for tourism (2013.03 IEEE, Kevin Meehan, Tom Lunney, Kevin Curran, Aiden McCaughey)
6	<ul style="list-style-type: none"> •
7	<ul style="list-style-type: none"> •
8	<ul style="list-style-type: none"> •
9	<ul style="list-style-type: none"> •
10	<ul style="list-style-type: none"> •