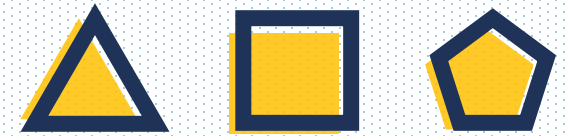




인스타그램 출구 별 추천 서비스

With 김유정(조장), 김상화, 박철민, 이기연

목차



1. 프로젝트 계획

- 추진목표
- 목적 및 필요성
- 기대효과
- 과제 수행 범위
- WBS, Gantt Chart

2. 요구사항 정의

- 요구사항 목록

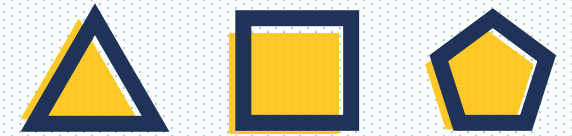
3. 시스템 구조 정의

- 프로세스 설계
- UI 구조 설계
- 화면 정의

4. 결과

- 코드 및 결과 화면

5. 후기



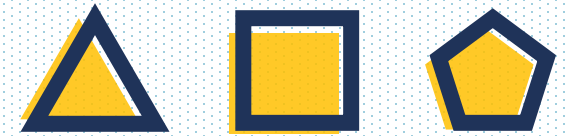
프로젝트 계획

추진목표



지역별 특징을 확인하고
추천을 위한 데이터 수집 및 분석

목적 및 필요성

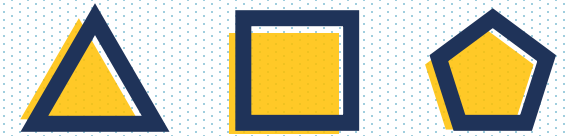


■ 인스타그램 선정이유

인스타그램은 한국에서 가장 많이 사용되는 SNS 중 하나입니다.

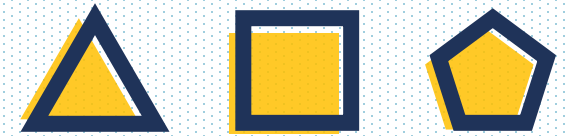
- 한국인의 SNS 사용률이 높고, 인스타그램은 그중 10~40대 이용자 수가 많음
 - 방문한 장소에서 즉각적으로 게시물을 올리고, 해시태그를 통해 정보를 제공함
-
- 첫 방문에도 원하는 곳을 쉽게 갈 수 있음

기대효과



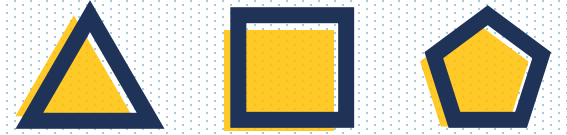
- 모란, 홍대, 판교의 정보를 수집하여 주로 어느 곳에 방문하는지 알 수 있음
- 선택의 폭을 줄여 시간을 절약할 수 있음

과제 수행 범위



- 웹 스크래핑
 - 인스타그램에서 글 내용, 해시태그 및 위치태그 가져옴
- 데이터 분석
 - 워드클라우드 시각화를 위한 데이터 정제
 - 스크래핑으로 가져온 데이터 분석하여 다양한 위치정보 구함
 - 카카오 API 사용해서 가게 위치 구함

과제 수행 범위



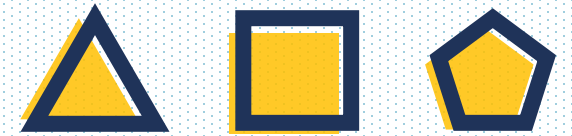
- 데이터 저장
 - 위치명, 좌표, 명사, 명사의 카운팅 값 저장 (CSV)
- 데이터 시각화
 - 워드클라우드를 통해 지역별 특징을 파악
 - folium을 사용해 가게 위치를 지도로 출력
 - 추출한 데이터의 통계 결과를 그래프로 나타냄

WBS (Gantt Chart)

계획

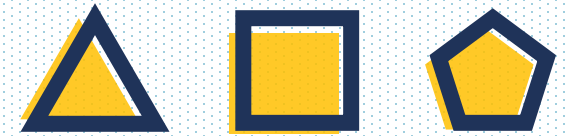
→ 실행

No.	TASK & ACTIVITY	DUE DAY		기간	상태	3월																4월								
		START	END			1W				2W				3W				4W				5W								
1	분석	03월 07일	03월 11일	3	완료	7	8	10	11	14	15	16	17	18	21	22	23	24	25	28	29	30	31	1	4	5	6	7	8	11
1.1	프로젝트 배경 및 시나리오 이	03월 07일	03월 07일	1	완료	→																								
1.2	개발환경 구축	03월 08일	03월 08일	1	완료		→																							
1.3	요구사항 정의	03월 08일	03월 11일	2	완료		→	→																						
2	설계	03월 14일	03월 18일	4	완료																									
2.1	시스템 구조 설계	03월 14일	03월 16일	2	완료				→																					
2.2	DB 설계	03월 17일	03월 18일	2	완료					→																				
3	구현 및 개발	03월 21일	03월 31일	13	완료																									
3.1	데이터 수집	03월 21일	3월 22일	8	완료								→	→	→	→	→	→	→	→	→	→	→	→	→	→	→	→	→	
3.2	데이터 정제	03월 23일	3월 24일	6	완료												→	→	→	→	→	→	→	→	→	→	→	→	→	
3.3	데이터 시각화	03월 25일	3월 30일	8	완료													→	→	→	→	→	→	→	→	→	→	→	→	
4	테스트	03월 31일	04월 01일	1	완료																									
4.1	통합 테스트	03월 31일	04월 01일	1	완료																				→	→	→	→	→	
5	Closing	04월 04일	04월 07일	2	완료																									
5.1	프로젝트 발표	04월 04일	04월 04일	1	완료																					→	→	→	→	
5.2	수정보완	04월 05일	04월 07일	1	완료																					→	→	→	→	
5.3	최종 보고서 발표	04월 07일	04월 07일	1	완료																					→	→	→	→	

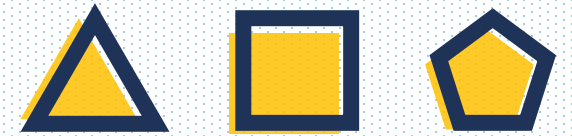


요구사항 정의

요구사항 목록

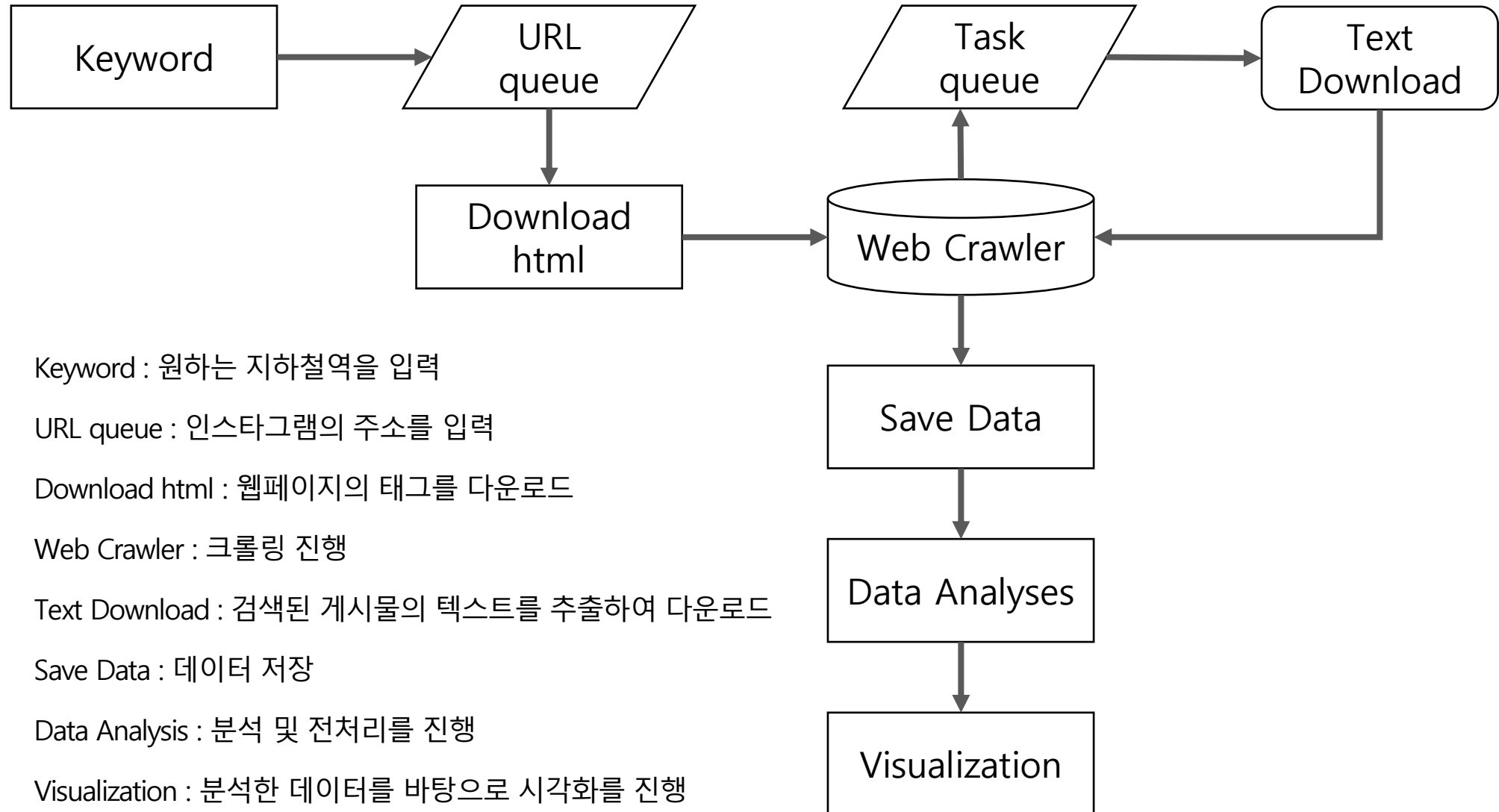


- 쿠키 방식 사용하여 로그인 진행
- 모란, 판교, 홍대 키워드 입력 후 검색
- selenium을 이용하여 글 내용, 해시태그 및 위치태그 데이터 추출(약 900개)
- 추출한 데이터를 csv 형식으로 저장
- 결과를 한눈에 파악할 수 있도록 정제 및 시각화

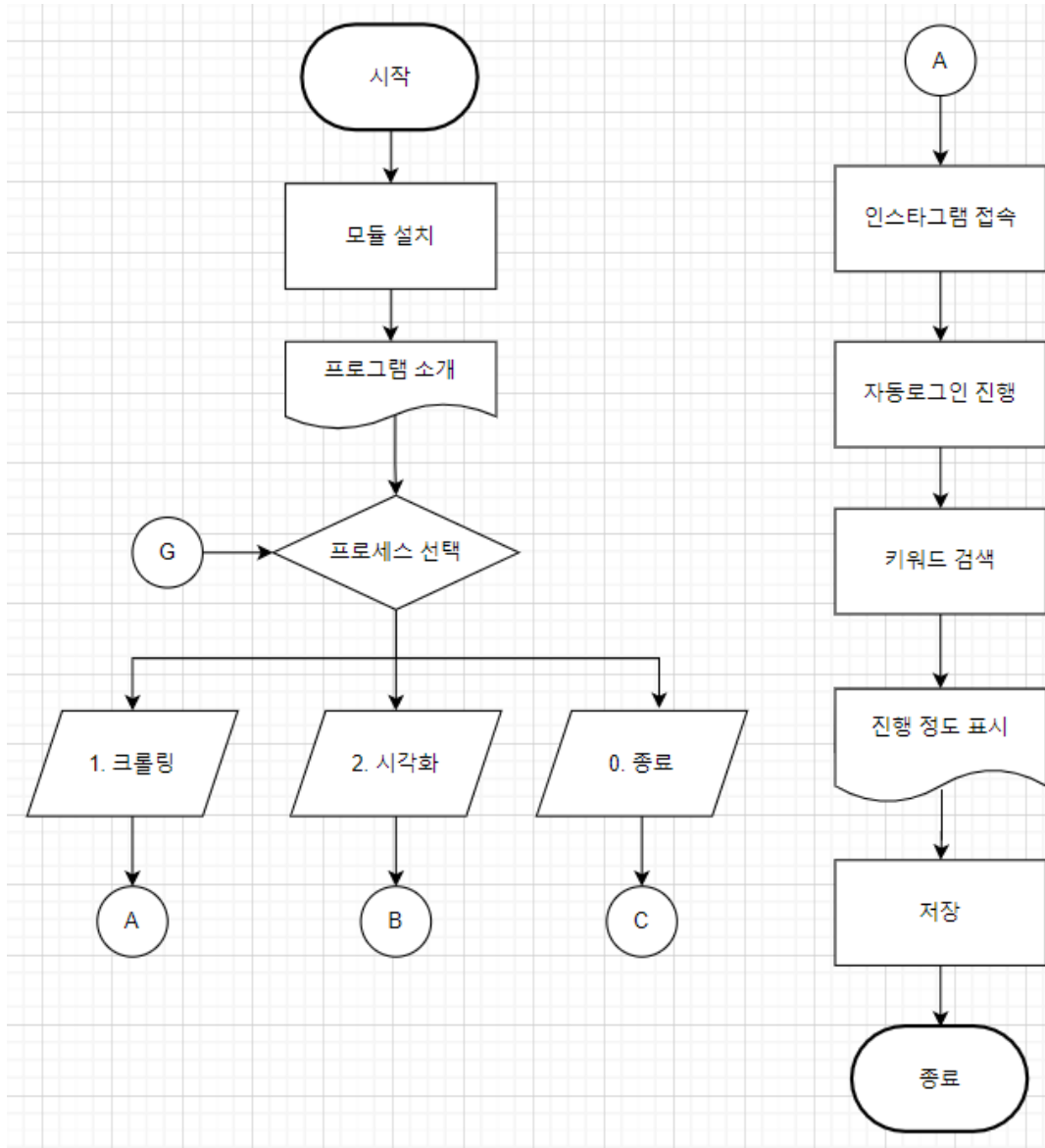


시스템 구조 정의

프로세스 설계



UI 구조설계



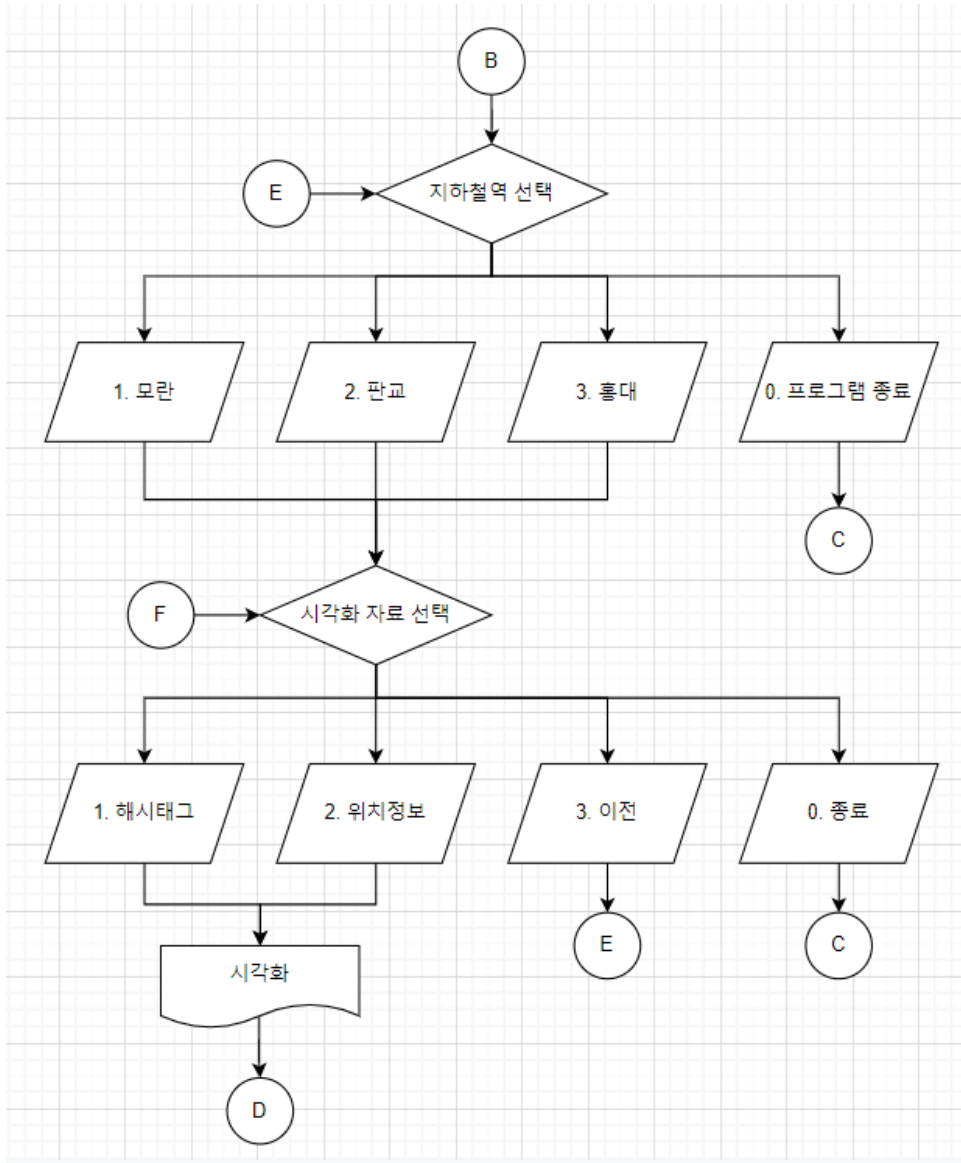
A : 인스타그램 크롤링 진행

B : 시각화 진행

C : 프로그램 종료

G : 시각화 진행 후 중단 선택 시

UI 구조설계



B : 시각화 진행

C : 프로그램 종료

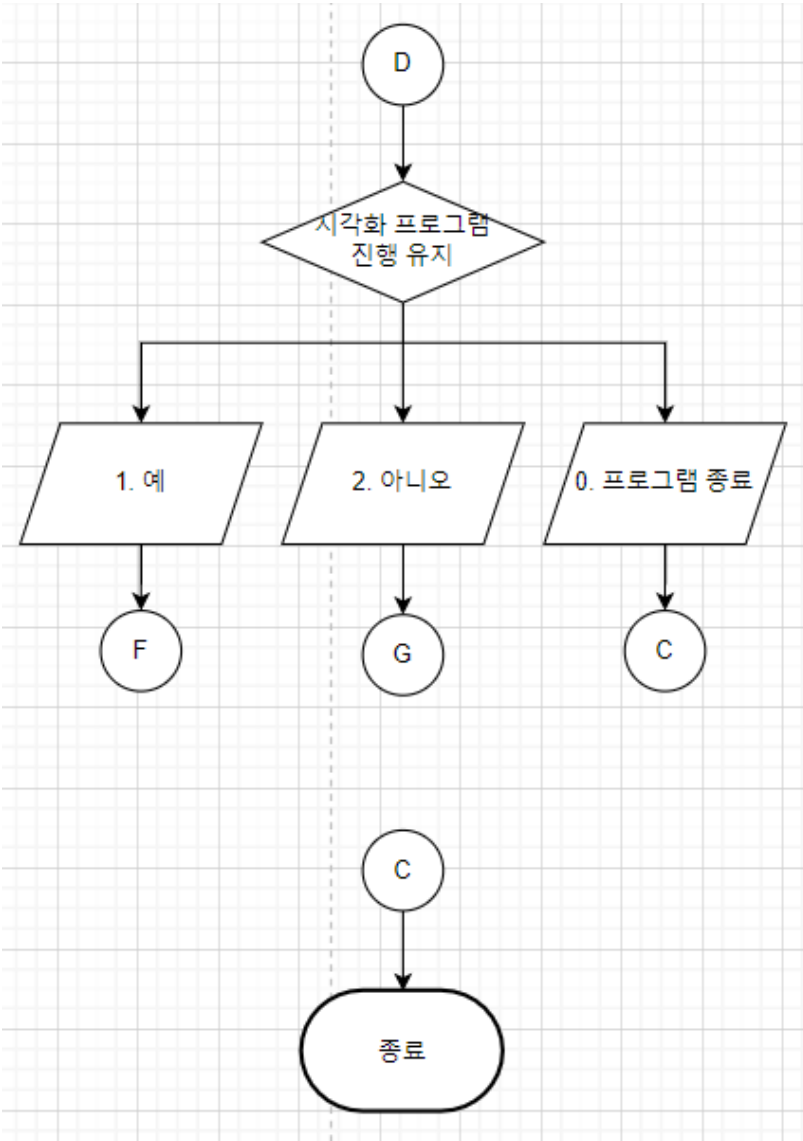
D : 해시태그 및 위치정보 시각화

(막대그래프, 워드클라우드 또는 지도)

E : 이전 메뉴로 이동, 지하철역 재선택

F : 시각화 프로그램 진행 유지를 선택한 경우

UI 구조설계



C : 프로그램 종료

D : 해시태그 및 위치정보 시각화

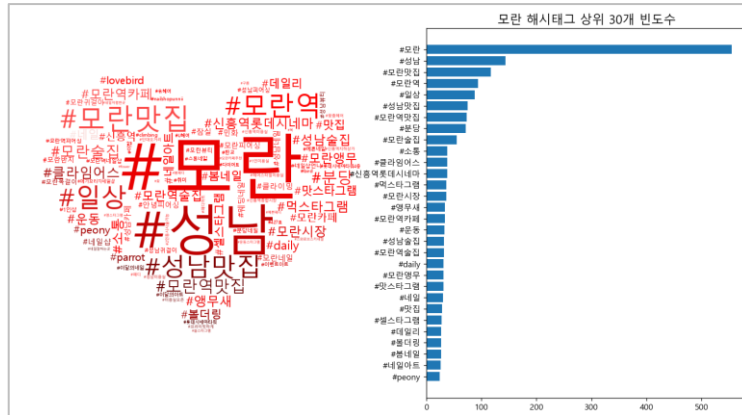
(막대그래프, 워드클라우드 또는 지도)

F : 시각화 프로그램 진행 유지를 선택한 경우

G : 시각화 프로그램 진행 유지를 선택하지
않은 경우(프로세스 선택으로 이동)

화면 정의

1. 해시태그 정보



[Step1]

원하시는 프로세스를 선택해주세요.

1. 크롤링을 진행
2. 시각화를 진행
0. 종료

입력 : 1 -> 크롤링 진행 -> 종료, 2 -> [Step2]

[Step2]

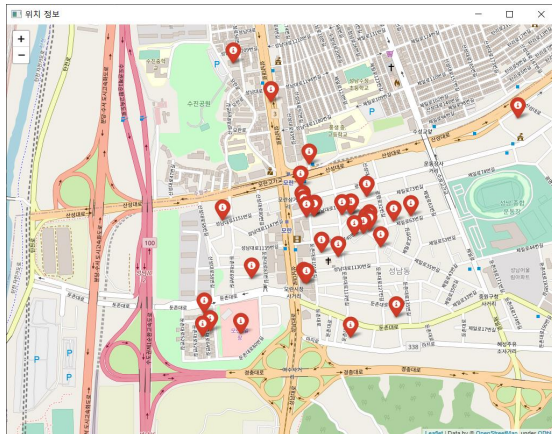
원하시는 지하철역을 선택해주세요.

(현재는 모란, 판교, 홍대만 선택 가능합니다.)

1. 모란
2. 판교
3. 홍대
0. 프로그램 종료

입력 : 모란 / 1,2,3 -> [Step3]

2. 위치 정보



[Step3]

모란 키워드에 대한 검색 결과물 중 시각화하고 싶은 자료를 선택해주세요.

1. 해시태그 정보
2. 위치 정보
3. 이전
0. 프로그램 종료

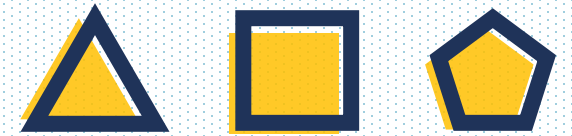
입력 : 1,2 -> [Step4], 3 -> [Step2]

[Step4]

00 키워드에 대한 시각화를 계속 진행하시겠습니까?

1. 예
2. 아니오
0. 프로그램 종료

입력 : 1 -> [Step3], 2 -> [Step1]



수행 결과

코드

외부모듈

```
'''
여러환경에서 사용될 수 있도록 프로젝트에 필요한 외부 모듈들을 설치합니다.
'''

프젝트 전형중 필요한 모듈을 추가 예정입니다.
'''

import sys
import subprocess

try:
    # 크롤링
    import bs4
    import selenium
    import webdriver_manager
    import pandas
    import tqdm

    # 시각화
    import matplotlib
    import wordcloud
    import folium

    # GUI
    import PyQt5
    from PyQt5 import QtWebEngineWidgets # GUI 화면진 사용을 위해 (추가 설치 필요)

    # 데이터 베이스 연동
    import pymysql

except:
    # pip 모듈 업그레이드
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "pip"])

    # 크롤링
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "BeautifulSoup4"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "selenium"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "webdriver_manager"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "pandas"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "tqdm"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "lxml"]) # 빠른 파싱을 위해

    # 시각화
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "matplotlib"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "wordcloud"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "folium"])

    # GUI
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "PyQt5"])
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "PyQtWebEngine"])

    # 데이터 베이스 연동
    subprocess.check_call([sys.executable, "-m", "pip", "install", "--upgrade", "pymysql"])
```

크롤링

```
'''
크롤링 테스트용 메인입니다.
'''

from crawler_pkg import * # 크롤링, 시각화 기능을 제공하는 패키지입니다.
# 멀티프로세싱을 위한 모듈
# multiprocessing의 pool은 top-level method에서만 사용할 수 있습니다.
from multiprocessing import Pool

# 사용자 정보를 자동화 함
# 전달 인수: query_list - 정의어 목록
def automate_user_data(user_query_list):
    # 사용자 입력정보
    tester_id_list = ["Crawler_test-With", "Crawler_test2-With", "Crawler_test3-With"]
    tester_pw_list = ["sugar0202", "sugar0202", "sugar0202"]

    # 가변 정보
    # insta_crawler1
    query_txt_list = ["모란", "관교", "홍대"]
    crawling_count_list = [900, 900, 900]

    selected_list = []

    for i in range(3):
        if query_txt_list[i] in user_query_list:
            save_file_name = f"insta_crawling_{query_txt_list[i]}_{crawling_count_list[i]}개.csv"
            insta_crawler = InstagramCrawler([tester_id_list[i], tester_pw_list[i],
            | query_txt_list[i], crawling_count_list[i], save_file_name])
            selected_list.append(insta_crawler)

    return selected_list

if __name__ == "__main__":

    query_txt_list = ["모란", "관교", "홍대"]
    object_list = automate_user_data(query_txt_list)

    # selenium은 다중 브라우저 테스트를 지원합니다.
    # 자동화된 방법 테스트를 수행합니다.
    # 크롤링 시간을 연속 할 수 있습니다.
    # 멀티 프로세스 (병렬처리)
    pocesse_count = len(query_txt_list)
    pool = Pool(processes = pocesse_count) # pocesse_count 개의 프로세스를 사용합니다
    pool.map(crawling_process, object_list)
```

시각화

```
'''
시각화 테스트용 메인입니다.
'''

# 화면화해서 시나리오로 구현 예정입니다.
'''

from crawler_pkg import *

# 사용자 입력 예외처리
def exception_input():
    while True:
        print("1. 원하는 키워드를 입력해주세요.")
        print("(문자는 모란, 관교, 홍대만 입력 가능합니다.)")
        keyword = input("입력 : ")

        ready_data = ["모란", "관교", "홍대"]

        if keyword.replace(" ", "") == "":
            print("공백어가 입력되지 않았습니다. 다시 입력해주세요.")
        elif keyword not in ready_data:
            print("자료가 준비되지 않았습니다. 다시 입력해주세요. (모란, 관교, 홍대)")
        else:
            break

    return keyword

# 시각화 시나리오 프로세스
def visualization_process():
    print("-" * 80)
    print("시나리오가 시작되었습니다.")
    print("시각화 데이터를 입력하면 주변 지역 서비스 정보를 보여드립니다 WITH입니다.")
    print("-" * 80)

    keyword = exception_input()
    print()

    print(f"2. {keyword} 키워드에 대한 검색 결과를 볼 시작과하고 싶은 자료를 선택해주세요.")
    print("(1. 위치 정보 2. 위치 정보)")
    choice = int(input("입력 : "))
    print()

    if choice == 1:
        print("-" * 80)
        print("시각화 데이터를 입력하면 주변 지역 서비스를 보여드리겠습니다.")
        print("시각화 데이터를 입력하면 주변 지역 서비스를 보여드리겠습니다.")
        print("-" * 80)
        visual_wordcloud_process(keyword)
    elif choice == 2:
        print("-" * 80)
        print("시각화 데이터를 입력하면 주변 지역 서비스를 보여드리겠습니다.")
        print("시각화 데이터를 입력하면 주변 지역 서비스를 보여드리겠습니다.")
        print("-" * 80)
        visual_map_process(keyword)
    else:
        pass

    print()

if __name__ == "__main__":

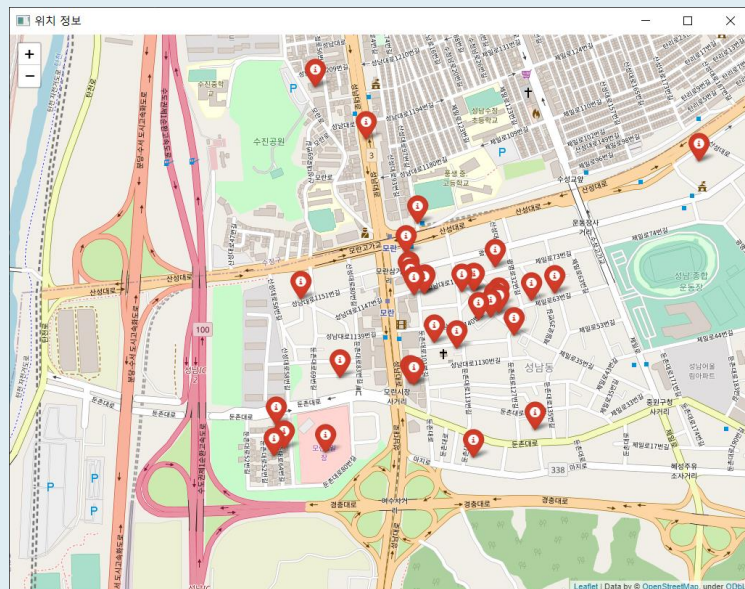
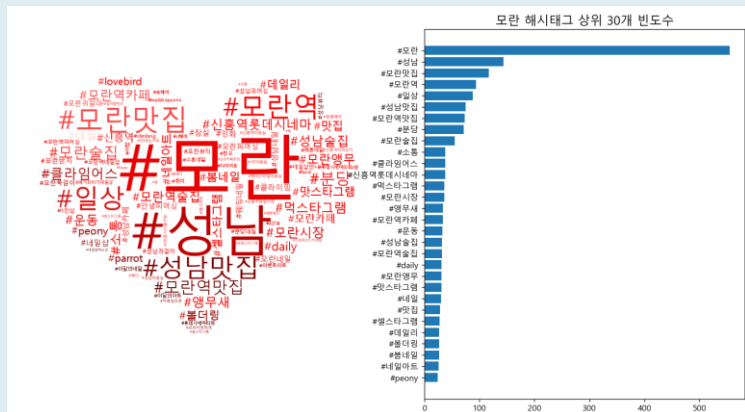
    while True:
        visualization_process()

        check_continue = int(input("3. 계속하시겠습니까? [ 1. 예 2. 아니오 ] : "))
        print()

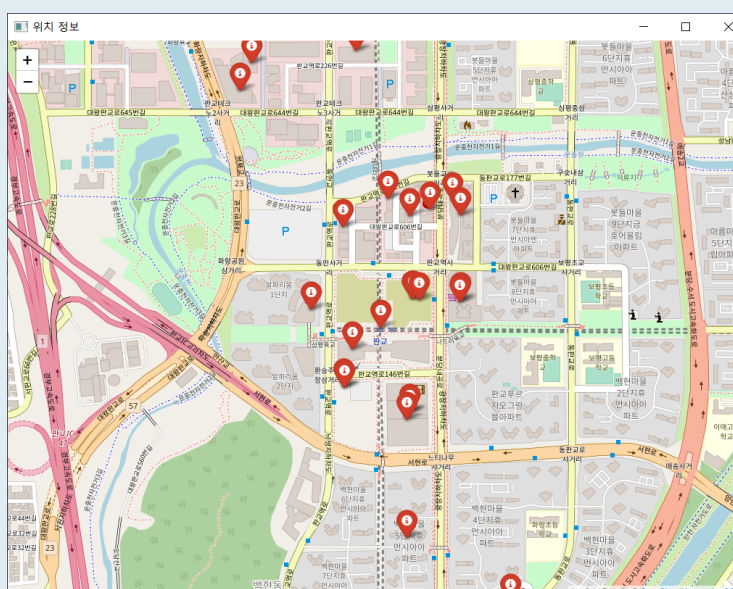
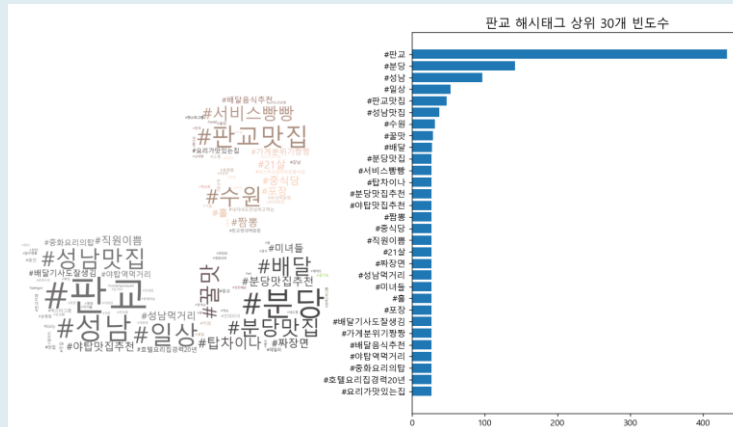
        if check_continue == 2:
            print("4. 종료합니다.")
            break
        else:
            pass
```

결과 화면

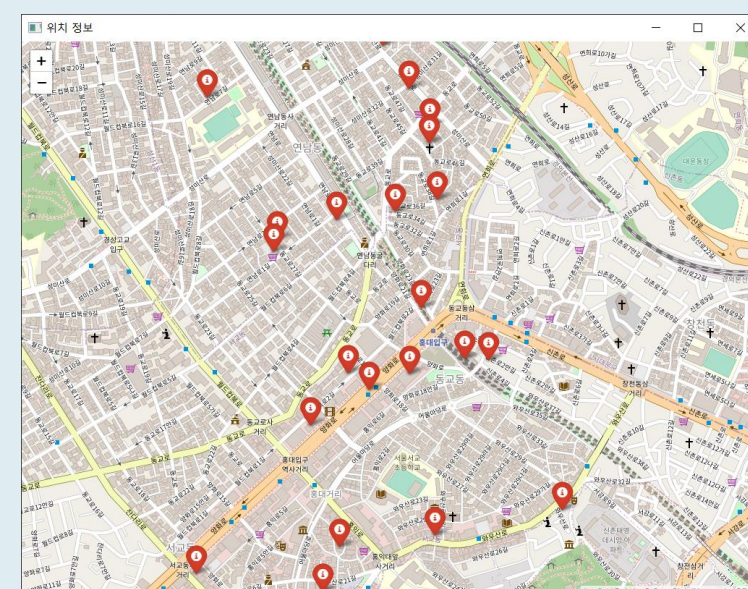
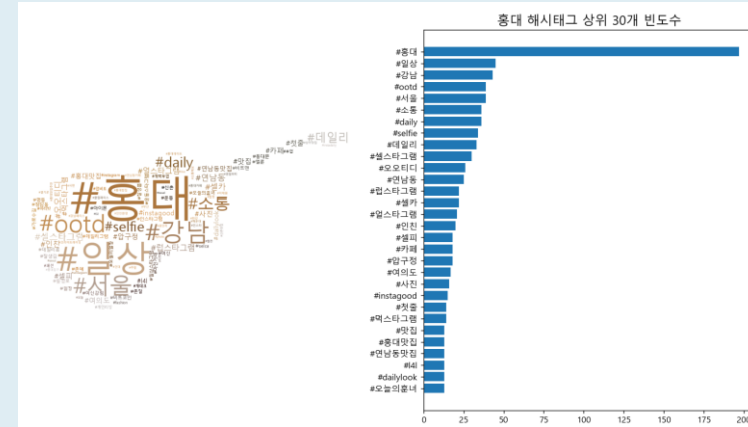
모란



판교



홍대



후기

김유정	사람들이 많을수록 의견을 모으는데 시간이 걸리지만 더 좋은 방향으로 나아갈 수 있는 힘이 된다는 걸 알 수 있었습니다. 또한 이론으로 배웠던 지식이 실제로 사용되는 모습을 보면서 공부한 보람을 느낄 수 있었고, 다른 사람들의 코드를 보면서 많이 성장할 수 있었습니다.
김상화	예전에 했던 개발일이 그리워지는 순간이었으며, 머리속에서 그리던 것들이 아직은 손가락이 따라오지 못하고 있지만 조금씩 나아가는 모습에 나에게 칭찬을 해주고 싶었습니다.
박철민	이번 프로젝트를 통해 데이터를 수집하고, 시각화해 보면서 여러 외부 모듈들을 활용하고 익히는데 유익한 시간이 되었습니다. 의미 있는 결과물을 얻기 위해서는 데이터 수집, 데이터 정제 등 모든 과정에서 데이터 분석에 많은 노력이 필요하다는 걸 깨닫게 되었습니다.
이기연	프로젝트를 하며 다양한 방법들이 있는 것을 알게 되었습니다. 아직 부족한 방법들을 공부 하며 팀원들과 소통하여 배우고 의견을 받으며 회의 하는 것의 중요성을 깨달았습니다.

감사합니다