2020년도 2학기

[종합설계프로젝트]

최종 보고서

주제 : Posting-Monitor

11조

2014313795 김성섭

2014310312 김하림

2014311062 이진욱

Github : https://github.com/harim-k/Posting-Monitor

I. 소개

Posting-Monitor는 웹사이트 모니터링 서비스입니다. 사용자가 등록한 웹사이트에 포스팅이 발생하면 메신저(카카오톡과 텔레그램 중 선택)으로 알려줍니다. 키워드 설정을 통해 관심 있는 포스팅에 대한 모니터링이 가능합니다.

예를 들면, 사용자가 Posting-Monitor에 카카오 채용공고 웹사이트(https://careers.kakao.com/jobs)와 "머신러닝" 키워드를 등록하면, 머신러닝 관련 새로운 포스팅이 올라올 때마다 알림을 받아볼 수 있습니다.

II. 관련 기술 조사

- 크롤링

1. 정의

크롤링(crawling)이란 조직적, 자동화된 방법으로 World Wide Web을 탐색하는 작업을 말합니다. Web상에 존재하는 Contents를 수집하거나, Selenium등 브라우저를 프로그래밍으로 조작하여 필요한 데이터만 추출하는 등등의 작업을 수행합니다.

2. 규제

크롤링에 대한 규제로 로봇 배제 표준(robots exclusion standard), 로봇 배제 프로토콜(robots exclusion protocol)이라고 불리는 Robot.txt가 있습니다. 웹사이트에 로봇이 접근하는 것을 방지하는 규약입니다.

이 규약은 권고안이며, 크롤러가 robots.txt 파일을 읽고 접근을 중지하는 것을 목적으로 합니다. 하지만 robots.txt를 무시하고 읽는 것 또한 가능합니다.

3. 방식

크롤링의 방식은 다음과 같습니다.

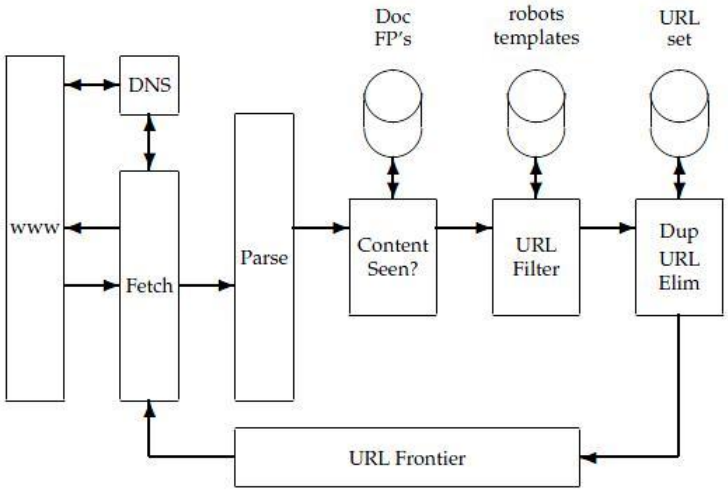


Figure 20.1 Basic crawler architecture.

Fetcher와 Parser, Frontier 세 부분으로 구성되며, 각각의 기능은 다음과 같습니다. Frontier는 Web을 탐색하며 url을 fetcher에 넘겨줍니다. Fetcher는 해당 url의 html을 적절히 처리를 하여 Parser로 넘겨줍니다. Parser는 최종적으로 html 내의 내용의 링크를 찾아 Frontier로 넘겨줍니다.

저희는 이를 간소화하여 web의 링크를 추적하는 크롤링을 구현했습니다.

- 기존 웹사이트 모니터링 서비스

1. Google Alerts

Google Alerts는 Google 검색 결과의 변화된 내용을 이메일로 전송합니다.

참조 : https://www.google.com/alerts

2. Uptime Robot

웹사이트를 5분마다 모니터링하여 이메일, 문자메시지, 트위터 등으로 알림을 받을 수 있습니다. 무료로 최대 50개의 사이트 모니터링 가능합니다.

참조 : https://uptimerobot.com/

- Posting-Monitor의 차별점

  1. **다수의 웹사이트**를 모니터링 가능합니다.
  2. **키워드**를 설정하여 원하는 포스팅만 모니터링 가능합니다.
  3. 따로 알림을 제공하지 않는 웹페이지를 모니터링 가능합니다.
  4. 접근성이 높은 메신저**(카카오톡 or 텔래그램)**으로 알려줍니다.

III. 사용 라이브러리 및 API

1. Python 라이브러리

   selenium : url 접속 및 html 추출

   BeautifulSoup : html 분석 및 href 추출

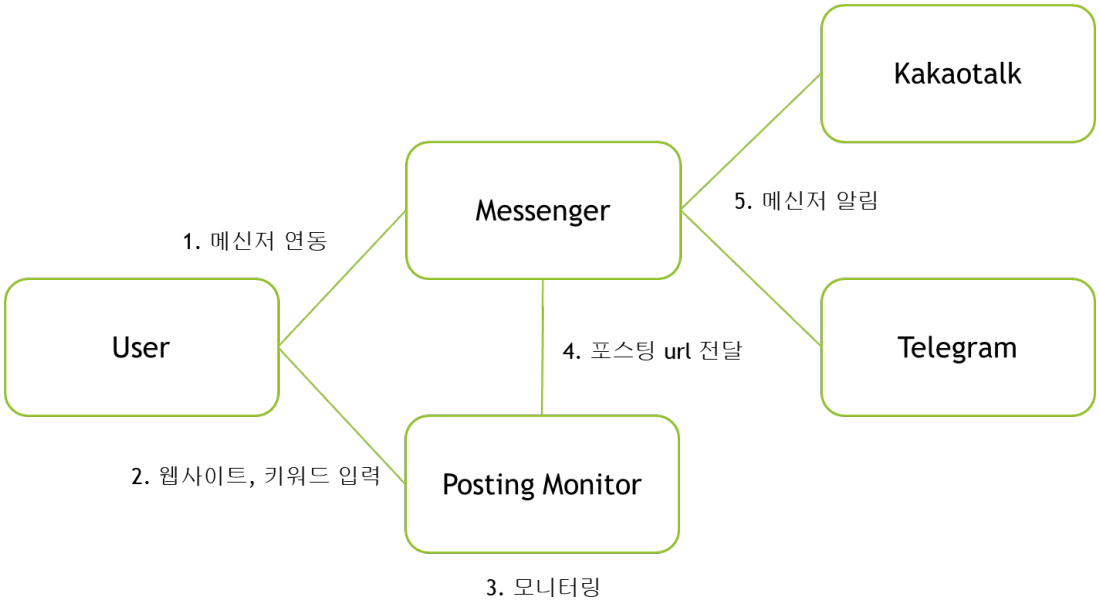
   tKinter : GUI 구현

2. 메신저 API

   카카오톡 : 나와의 채팅

   텔레그램 : telegram-bot

IV. 구조



1. User(GUI)

   역할 : 사용자 인터페이스. 모니터링할 웹사이트와 알림을 받을 카카오톡 정보를 입력 받습니다.

   담당 : 이진욱

2. Posting Monitor

   역할 : 웹사이트를 모니터링합니다. 새로운 포스팅이 발생하면 링크를 Messenger에 전달합니다.

   담당 : 김하림

3. Messenger

   역할 : 사용자가 설정한 메신저를 연동하고, Posting Monitor로부터 받은 url을 메신저로 알립니다.

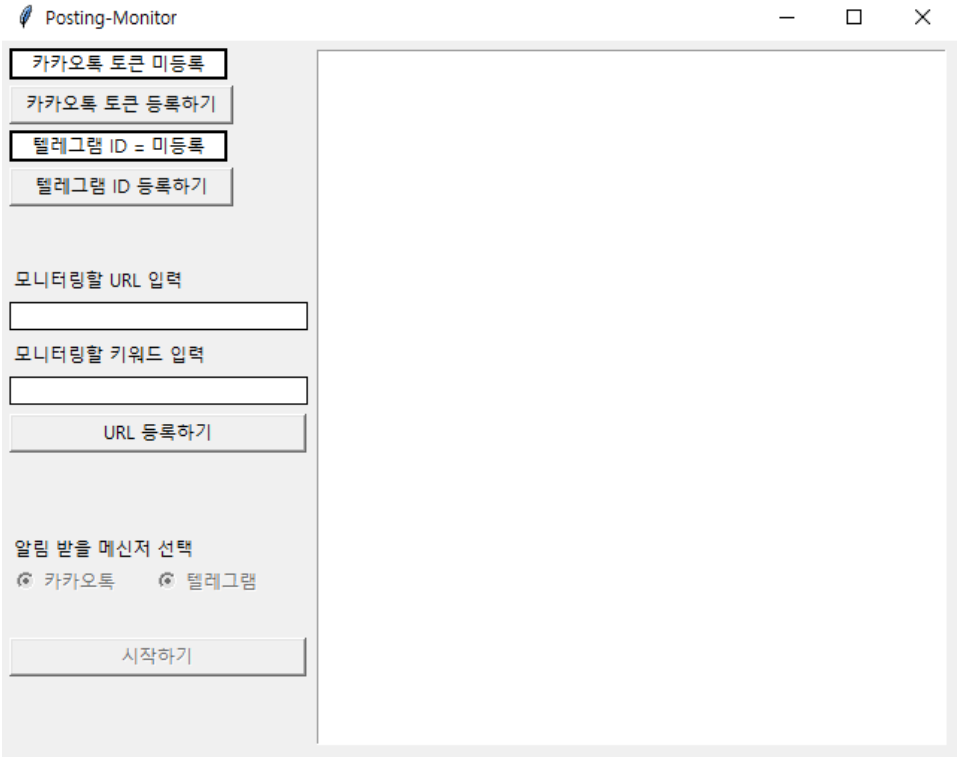
   담당 : 카카오톡 - 김성섭

   텔레그램 - 이진욱

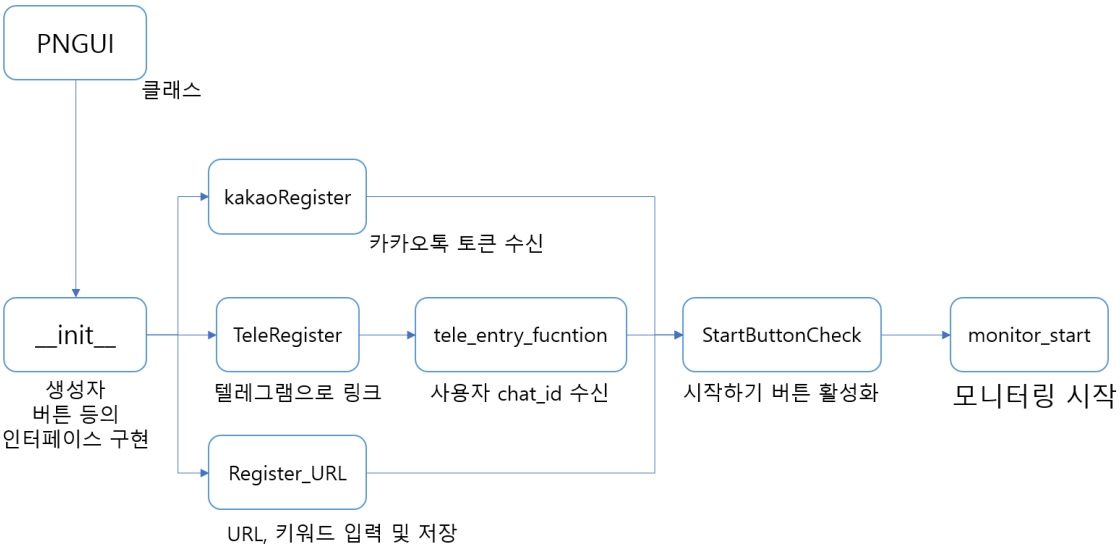
V. 구현

1. User(GUI)

GUI는 빠르고 직관적인 Python3의 내장 라이브러리인 Tkinter를 사용했습니다.



- GUI 구조



생성자에 여러 버튼, 레이블, 리스트박스 등의 인터페이스가 선언되어 있습니다. 사용자가 '카카오톡 토큰 등록하기' 버튼을 누르면 내부의 kakaoRegister 함수가 호출되고, '텔레그램 ID 등록하기' 버튼을 누르면 내부의 TeleRegister 함수가 호출됩니다. URL과 키워드를 입력하고 'URL 등록하기' 버튼을 누르면 리스트박스에 모니터링할 웹 사이트와 키워드를 저장합니다.

이후 알림을 받을 메신저를 선택하면 StartButtonCheck 함수를 호출해 시작하기 버튼이 활성화되고, '시작하기' 버튼을 누르면 monitor_start 함수가 호출되고 리스트박스 내 저장된 URL과 키워드를 이용해 모니터링을 시작합니다.

2. Posting Monitor

- 모니터링 방식

1. Selenium Webdriver를 통해 웹사이트에 접속하여 html 추출합니다.
2. BeautifulSoup을 통해 추출한 html를 분석하고 href(링크)를 추출합니다.

- 과정

A. 웹사이트 첫 접속

1. url에 접속하여 html 추출
2. html의 모든 href를 저장 (이후 새로운 href 확인을 위해)

B. 포스팅 체크

3. 일정 주기마다, url 접속하여 html 추출
4. html의 새로운(기존에 없던) href를 저장

C. 키워드 체크

5. 새로운 href에 접속하여 html 추출
6. html 내에 키워드가 존재할 경우, href를 Messenger에 전달

3. Messenger

Python에서 카카오톡 메시지 API와 텔래그램 봇 API를 활용하여 메시징 기능을 구현합니다.

1. 카카오톡

- 카카오톡 메시지 API

카카오톡 메시지 API는 다음과 같이 지원됩니다. 모바일 앱 형태로 안드로이드, iOS를 둘 다 지원하며 웹의 형태로는 REST API를 지원합니다. 그 중 저희는 REST API를 활용했으며 카카오 웹서버와의 통신으로 프로세스가 이루어집니다.

그리하여, 저희는 카카오톡 기본 메시지 API를 사용하는 형태로 카톡 수신을 구현하기로 하였고, 프로세스는 다음과 같이 요약할 수 있습니다.

사용자가 저희 서비스를 이용하기 희망하면, 사용자는 본인의 카카오톡 계정으로 저희가 등록한 카카오 앱에 가입을 합니다.

그 과정이 이루어지면, 앱의 api를 통하여 '나에게 메시지 보내기' 기능을 이용하여, 해당 사용자는 카카오톡 계정의 '나와의 채팅'을 통하여 Notifier가 제공하는 내용을 받을 수 있습니다.

- 카카오톡 메시지 API 구현 함수 : run_token_server()

토큰이 없을시 실행하면 되고, 페이지 연결을 실행하여 카카오 로그인을 한후 앱 가입, 메시지 수신허용을 하는 페이지를 실행합니다

이때 만약, 메시지 수신동의에 체크를 하지 않았다면 메시지 알림을 받기 불가능하므로 꼭 체크를 해야합니다. .

그리고 get 요청에 대응하기 위한 서버 또한 실행합니다.

로그인과 동시에 서버에서 get요청을 처리하여 인가코드를 카카오톡api 서버 에 post요청을 보내어 토큰을 발급받아옵니다. 로컬에 저장을 한 후, 해당 토큰의 이름을 반환합니다. 또한 redirect url에 토큰 발급이 완료되었음을 알려줍니다.

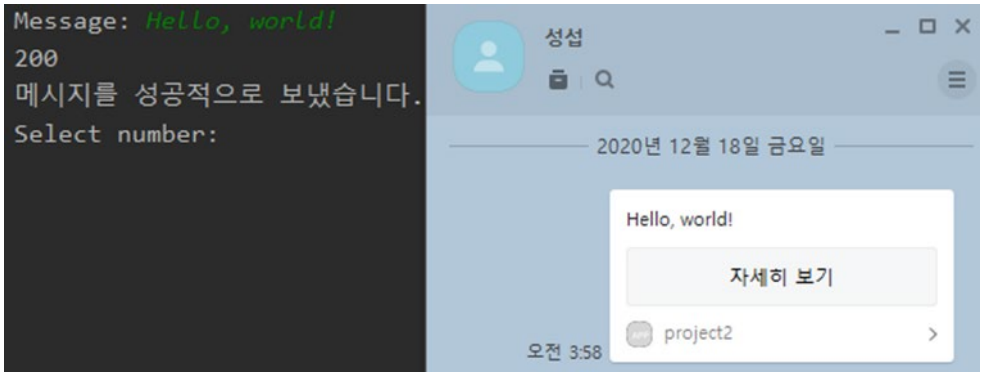
Run_token_server 함수 안에서 post요청을 보내는 과정입니다. 앱의 id와 어디로 redirect시킨지를 알 수 있는 redirect_uri, 인가코드가 post의 data부분에 첨부됩니다.

토큰은 통상 6시간의 만료기간을 지니며, refresh token을 통하여 다시 업데이트 할 수 있습니다. 서버의 refresher thread는 token refresh를 만료직전에 수행합니다. 결론적으로 토큰을 기반으로 앱에 등록된 카카오톡 사용자에게 메시지를 발송, 그 외 몇 가지 행동을 할 수 있습니다.

-카카오톡 메시지 API 구현 함수 : sendMsg(tokenName, msg)

tokenName을 인자로 받아 로컬에 저장된 토큰을 불러와 post의 header에 액세스 토큰을 첨부합니다. 두번째 인자로 받은 msg는 data의 text항목에 넣어줌으로써 메시지가 발송될 수 있게 하고, 최종적으로 카카오톡 api서버에 post요청을 발송합니다.

최종적으로 발송이 완료된 모습은 다음과 같습니다.



2. 텔레그램

- 텔레그램을 이용한 알림 보내기

사용자가 텔레그램에서 메시지로 알림을 받게 하기 위해 텔레그램 봇 API를 사용하기로 했습니다.

봇이 사용자에게 텔레그램 메시지를 보내려면

1. 사용자가 봇에게 메시지를 최소 1회 이상 보내야 하고,

2. 봇이 사용자의 chat_id를 알아야만 합니다.

일단 사용자가 먼저 봇에게 메시지를 보낸다면 텔레그램 봇의 기능을 이용해 정보를 얻어올 수 있고, chat_id를 얻어온 이후 직접적인 웹 요청으로 봇이 메시지를 보내게 됩니다.

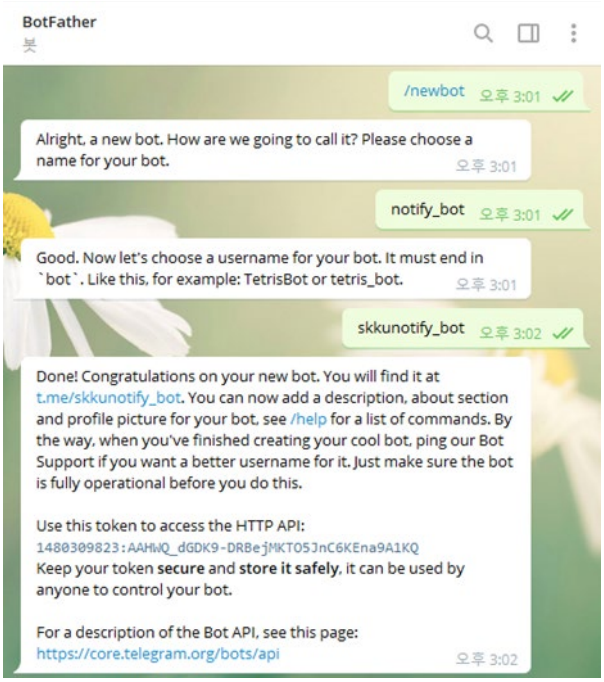


그림: BotFather를 이용한 간단한 봇 생성과 토큰 발급

- 텔레그램 사용자 등록용 함수: tele_register_v2_auto(user_id)

웹 브라우저를 열어 봇에 연결되는 링크를 열고, 사용자가 봇에 메시지를 보냈는지 확인한 이후 update 객체의 정보를 인자로 들어온 user_id와 비교해 사용자의 chat_id를 return하는 함수입니다.

- 텔레그램 메시지 송신 함수: tele_sendmsg(cid, msg)

사용차의 chat_id를 인자로(cid), 메시지를 인자로(msg) 받고 직접적인 웹 요청을 통해 봇이 사용자에게 메시지를 보내도록 하는 함수입니다.

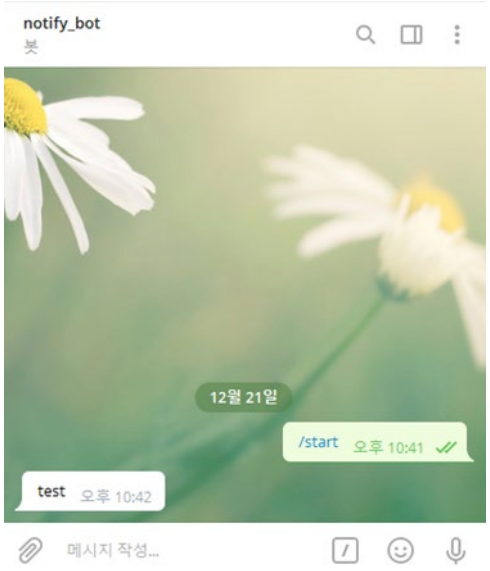


그림: 위 함수들을 이용해 봇이 사용자에게 메시지를 보내도록 한 모습.

VI. 결과 및 분석

- 테스트 웹사이트

1. 에브리타임 인사캠 자유게시판
2. 트위터
3. saramin(사람인) 검색
4. youtube 채널
5. 카카오 채용 공고
6. 성균관대학교 소프트웨어학과 공지사항
7. 디씨인사이드 리그 오브 레전드 갤러리
8. 서울주택도시공사 주택임대 공지사항

- 모니터링 가능 웹사이트

1. 에브리타임 인사캠 자유게시판
2. 트위터
3. youtube 채널

- 모니터링 가능 추정 웹사이트

직접 포스팅이 불가하여 검증은 되지 않았지만, html 분석 결과 가능할 것으로 추정되는 웹사이트입니다.

1. 카카오 채용 공고
2. saramin(사람인)

- 모니터링 불가능 웹사이트

1. 서울주택도시공사 주택임대 공지사항
2. 성균관대학교 소프트웨어학과 공지사항

    이유 : 다른 유형의 href

    href = “https://cs.skku.edu/news/recent/list#”

3. 디씨인사이드 리그 오브 레전드 갤러리

    이유 : 동적 url

    url에 page number가 포함되어 있어, 시간이 지나면 url이 바뀝니다.

https://gall.dcinside.com/board/view/?id=leagueoflegends4&no=1008669&page=1

VII. 참고문헌

[1] robot.txt, http://www.robotstxt.org