**AdBlogBlock Development Process**

사용 언어: 파이썬 -> 자바스크립트로 포팅하여 크롬 확장 프로그램으로 개발 예정

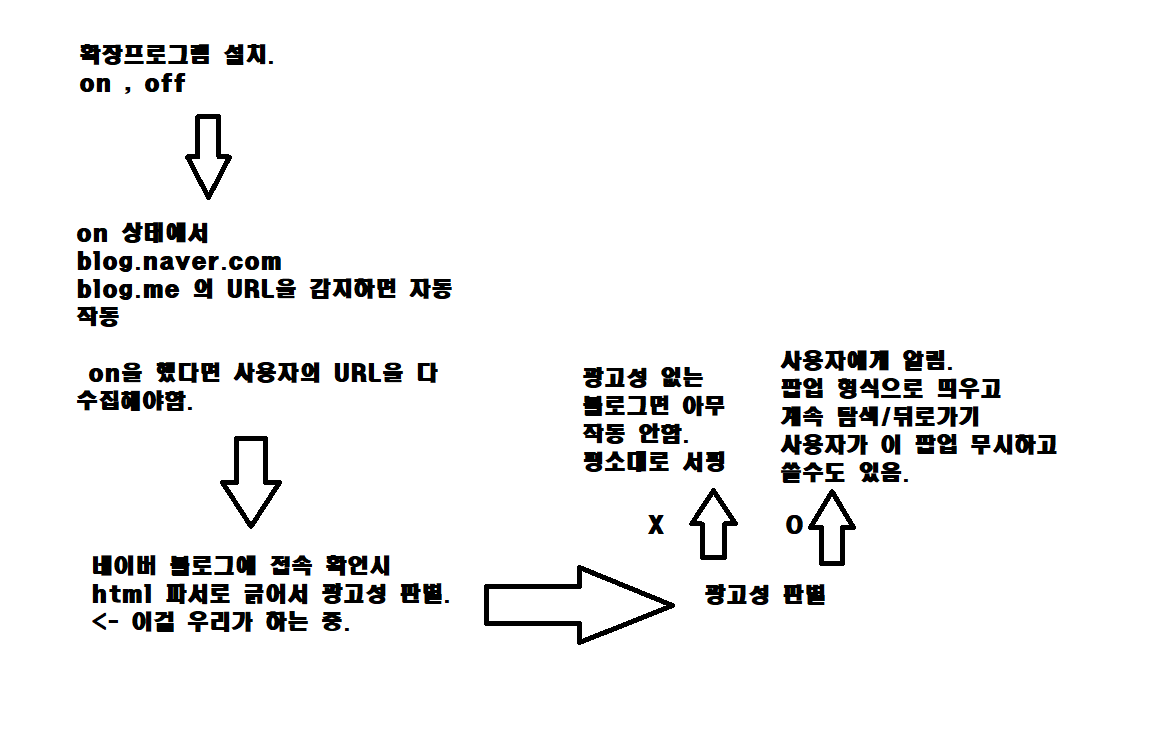
개발 일정

|  |  |  |
| --- | --- | --- |
| 3.24 | 파싱 모듈 개발 | 개발 계획 수립 |
| 3.31 | 광고성 판별 알고리즘 구현 |
| 4.7 |
| 4.14 | 중간고사 BREAK | |
| 4.21 |
| 4.28 | 파싱 모듈 테스트 및 보강 | 광고성 판별 알고리즘 구현 |
| 5.5 | 광고성 판별 모듈 개발 | 테스트 케이스 수집 |
| 5.12 |
| 5.19 | 광고성 측정 모듈 테스트 | 크롬 확장 프로그램 개발 & 예비 주간 |
| 5.26 | 통합&테스트 |
| 6.2 | 크롬 확장 프로그램으로 포팅& 최종 테스트 | |
| 6.9 | 기말고사 BREAK | |
| 6.16 |
| 6.23 | 유지보수 및 배포 | 예비 주간 |
| 개발 종료 | | |

각 Phase 별 상세 설명:

* 파싱 모듈 개발: 주어진 html을 파싱 하는 모듈. 본문은 물론, 사진, 스티커 까지 모두 파싱이 가능해야 함. 스티커의 종류까지 구분할 수 있으면 금상첨화. 기타 무엇이 필요한지도 메모한 후 구현해야 함
* 파싱 모듈 테스트 및 보강: 몇 개의 html 문서를 가져와서 파싱이 제대로 이루어지나 테스트 해봄. 간단한 테스트용 모듈 제작도 필요함. 그냥 html 불러와서 파싱모듈 불러서 파싱하고 그 결과를 파일로 다시 내보내주는 정도면 충분할 듯.
* 광고성 판별 모듈 개발: 광고성 판별 알고리즘을 토대로 실제로 코드화 시키는 작업.
* 광고성 측정 모듈 테스트: 파싱한 내용들을 따로 넣어서 테스트. 역시 테스트 전용으로 간단한 모듈을 하나 만든 후 제대로 작동하나 테스트
* 광고성 판별 알고리즘 구현: 이 프로그램의 알파이자 오메가. 해당 게시글이 광고글임을 판별할 수 있는 알고리즘을 구현한다. 여러가지 요소들을 종합적으로 판단할 수 있는 알고리즘이여야 하며, 카톡방에서 브레인스토밍한 내용을 총망라해서 구현한다. 이것도 테스트할 수 있으면 좋음.
* 테스트 케이스 수집: 광고글들을 링크의 형태로 수집. 최대한 다양한 종류의 광고글들을 넣어야 하며, 일부는 광고글과 비슷하지만 아닌 걸로 모은다.
* 통합&테스트: 개발한 두 모듈을 하나로 통합하고, 이 형태로(python 응용 프로그램) 동작하나 테스트.
* 크롬 확장 프로그램 개발: 크롬 확장 프로그램으로 배포 예정이니, 위에서 말한 모듈을 바로 넣을 수 있도록 manifesto 등 기본 베이스만 미리 개발해 놓음. 또한 이 기간에 Javascript를 미리 조금 공부해서 나중에 지장이 없도록 함.
* 크롬 확장 프로그램으로 포팅& 최종 테스트: 확장 프로그램으로 포팅 후 최종 테스트.
* 유지보수 및 배포: 스토어에 올려서 실제 배포.

Program 작동 알고리즘:



프로그램의 기술적 로직:

chrome API or Javascript 통해 현재 URL 읽어오기

파싱 모듈로 URL 전달

사진&텍스트 따로 파싱

스티커와 사진 평가

텍스트: 광고성 문자열 평가

광고성 판별 알고리즘 작동

프로그램 작동(Pass or Block)

Use Case

1. 사용자가 평범한 네이버 뉴스나 중앙대학교 홈페이지에 들어간다.

ABB를 On으로 해 두어도 아무 일도 일어나지 않는다.

1. 사용자가 “이어폰 추천” 이라고 네이버에 검색 후 어떤 블로그에 들어간다.

타겟 URL에 blog.naver.com/????? 식으로 있을 거고, blog.naver.com에 반응하여 작동한다.

(물론 ABB는 On 상태이다.) 내부에서 광고성을 판별한 결과 광고 블로그가 아니라 결론을 내렸다. 따라서 아무런 작용도 하지 않는다.

1. 사용자가 “흑석 미용실 추천”이라고 네이버에 검색하고 블로그에 들어간다.

타겟 URL에 blog.naver.com/???? 의 형식이 있어서 ABB가 작동한다.

내부에서 광고성 판별 결과 광고 블로그라고 결론이 내려졌다. 확장 프로그램의 팝업으로 “이 블로그는 광고성 블로그로 판단됩니다. 계속 보시겠습니까? [계속 보기] [나가기] “의 창이 나타난다. 사용자는 [계속 보기]를 누를 경우 그대로 블로그 문서를 읽을 수 있고, [나가기]를 누를 경우 이 블로그를 나간다. 기본적으로 “뒤로가기”의 기능을 하지만, 새 창으로 열어서 뒤로 갈 창이 없을 경우 창을 닫아버린다.

UI

