

Probe 개발 문서

기획 의도

기존에 있는 검색 사이트의 목적은 검색어를 포함한 정보를 많이 보여주는 데에 있습니다. 그래서 오히려 사용자가 원하는 정보가 명확할 경우 이런 특성이 방해가 됩니다. 예를들어 구글은 검색 도구를 통해 언어, 날짜, 포함여부(모든 결과, 완전일치)를 지정할 수 있지만 대신 정확도가 떨어 집니다. 네이버도 검색 옵션을 지원하지만 다른 웹사이트에서 가져오는 글은 마찬가지로 정확도가 떨어지며, 홍보성 글 때문에 오히려 불필요한 정보만 얻게 됩니다.

이처럼 사용자가 원하는 정보가 명확한데도 불구하고 다시 검색해보거나 사용자가 직접 해당 사이트를 들어가서 보는 게 나은 경우가 많습니다. 이러한 사이트가 여러 개가 되면 검색에 낭비되는 시간은 점점 늘어나게 됩니다. 이처럼 특정 정보를 원할 때 소비하는 시간을 줄이고자 한 앱입니다.

기능

1. 선호하는 키워드, 선호하는 사이트 조합을 등록한 후 사용자가 원할 때 페이지 단위로 검색(단위는 사용자가 설정).
2. 검색 후 원하는 페이지 단위로 다시 검색하거나 해당 결과물의 링크로 이동
3. 등록된 선호 조합들을 사용자가 설정한 시간(분)마다 백그라운드에서 검색하고 결과물을 가져옴. 이전에 검색된 결과물보다 최신 글이 있을 경우 알림
4. 링크로 이동했던 결과물들은 검색 기록에 보관

구현 계획

디자인

- Single Activity 구조, Material design을 사용해서 디자인
- 화면 구성은 안드로이드 Navigation component를 이용해서 시각적으로도 볼 수 있고

xml로 되어있어 구현에 필요한 자바 코드를 줄임

메뉴

- Toolbar와 BottomNavigationView를 이용해 각 Fragment로 이동
- Toolbar의 버튼은 뒤로가기, 사이트 추가, 데이터 정렬버튼으로 구성
- BottomNavigationView 메뉴는 홈, 검색기록, 설정탭으로 구성

검색

- html 파싱 라이브러리인 jsoup을 사용해 웹 크롤링 기능 구현
- 서버 구현이 가능할 경우 서버에서 크롤링을 처리한 후 데이터만 클라이언트에 전송
- AlarmManager, Service를 통해 백그라운드 작업 구현
- 페이지 단위는 Preference를 이용해서 저장
- 선호 사이트, 선호 키워드는 json으로 저장해서 앱이 꺼진 후 다시 실행되어도 유지
- 사이트의 글이 이미지가 포함되어 있을 경우 Glide 라이브러리를 이용해 이미지 처리
- 검색 결과물은 WebView로 연결해서 보여주고 이때 Toolbar와 BottomNavigationView는 원활한 탐색을 위해 숨김

한계와 해결 방안

선호 사이트가 자바 코드상으로 직접 구현(하드코딩)되어 있어 사용자가 직접 추가하기가 어렵습니다. 하지만 해당 앱의 소스가 공개되어 있고, 데이터 추가를 원하는 사이트의 html 구조를 알 경우에 기존의 코드를 참고해서 10~20줄 정도의 코드만 contribute를 해주시면 추가가 가능합니다. 또 네트워크 연결이 불안정할 경우 데이터를 제대로 읽어 들이지 못할 수 있는데, 이를 최소화 하기 위해 서버를 통해 필요한 정보만 주고 받게 할 수 있습니다. 하지만 캠프의 개발 기간 동안 서버는 구현하지 못했고, 향후에 지속적으로 개발할 예정입니다.

개발에 참고할 사이트

1) 국방부오픈소스아카데미

Osam.kr

2) jsoup

<https://blog.naver.com/cosmosjs/221294678259>

3) 바텀내비게이션뷰 예제

<https://dev-imaec.tistory.com/11>

4) AlertDialog 예제 ()

<https://webnautes.tistory.com/1094>

5) Material design

<https://devyurim.github.io/android/android/2018/01/17/android-3.html>

6) AsyncTask 사용법

<https://webnautes.tistory.com/1082>

<https://stackoverflow.com/questions/16752073/how-do-i-return-a-boolean-from-async-task>

7) Web crawling for multiple websites with different structures

<https://stackoverflow.com/questions/43712602/web-crawling-for-multiple-websites-with-different-structures>

8) jsoup 추출방법

<https://sehoonkim.tistory.com/158>

9) fragment 간 이동은 android jetpack navigation

<https://developer.android.com/guide/navigation/navigation-getting-started>

10) android jetpack

<https://developer.android.com/jetpack>

11) Glide 라이브러리

<https://github.com/bumptech/glide>

<https://youngest-programming.tistory.com/25>