**공동체 내에서 발생한 문제를 트랙킹하여 공론화해주는 어플**

문제 정의 : 한 때 핫했던 주제들. 심각한 문제라고 생각했음에도 불구하고 시간이 지나서 해결되었는지도 모르고 그냥 소리소문 없이 잊혀진 사안들이 존재한다. 문제가 미해결된 채 누적된다면 해당 공동체에서 같은 문제가 반복되기 마련이고 썩을 수 밖에 없다.

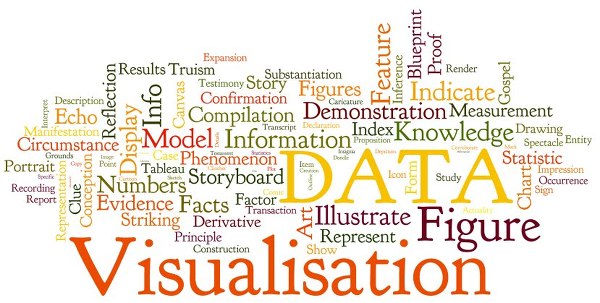
문제 원인 : 떠올랐던 사안을 정리하고[잊혀지지 않게], 해당 문제가 해결되었는지 끝까지 트랙킹하는 주체 세력이 없기 때문이다.

해결 : 우리 조는 핫한 주제들이 쉽게 잊혀지지 않게 정리하고, 이들이 해결될 때까지 트랙킹하여 미해결된 채 넘어가는 것을 방지하는 어플을 만들고자 한다.

기대효과 : 이 어플은 공동체가 보다 더 성숙해지는데 기여할 것이다.

--------------------------------

전체적인 프로세스 구상도

1. 데이터 시각화
   1. 데이터 가져오기 [jsoup]
   2. 저장 [DB]
   3. 시각화 [워드클라우드, d3.js 등]
2. 각 주제별 토론공간 만들기 [구성원의 공동체의 문제이해를 도움]
   1. 주제별 반응 정리

주제별 반응이 뜨거든 게시글들을 보여준다

* 1. 토큰 활용

토큰 활용을 통해 건전한 토론환경을 조성한다

1. 문제 해결 촉구

- 일정 양 이상의 관심을 가지고 있는 주제는 문제 책임자 및 관계처에 해결을 촉구할 수 있게 해준다.

- 여론 정리

1. 트랙킹

관계처의 답변을 기다리고, 이행 여부를 감시한다. 해결되지 않은채 시간이 흐르면 아직 해결되지 않았음을 구성원에게 공지한다.

참고

Point :

* 구성원들이 관심을 갖고 있는 주제를 **정리**한다.
* 주제들이 한눈에 들어오게 해야한다 = **데이터 시각화**
* 기존의 서비스와 차이점을 명확히 한다
* 효율적인 토론의 장을 만든다.
* 문제 해결 **이행 여부를 감시**한다.

**프로젝트 키워드 : 이슈 정리하여 시각화, 이행 여부 감시**

**9.30 회의 내용 정리**

지난 회의시간에…

문제를 인지하고, 그에 대한 대안의 틀은 잡았다.

대안의 전체적인 프로세스를 그려봤으며, 각 프로세스에 필요한 기술들을 알았다.

[크롤링, 데이터베이스, 워드클라우드, 어플리케이션 제작 ]

[금일]

공부해야할 양이 방대하여 분업을 해야하는데, 연결된 프로세스를 쪼개어 역할을 배정하기 위해선 우선 프로세스에 대한 이해가 있어야함을 인지함. 하지만 팀원 모두에게 생소한 분야이기 때문에, 기술에 대한 지식 습득의 필요성을 절감함

하지만 ( 크롤링,데이터베이스,워드클라우드 )과 (어플리케이션 제작)은 특성이 상이해 분업이 가능하다고 생각했으며 그래서 현재 단계에서 나눌 수 있는 분업은

1)크롤링,데이터베이스,워드클라우드 이해하기

2)어플리케이션 제작 이해하기.

4명이 1)을 맡고, 1명이(정지원) 2)를 맡는다.

1)이 베이스가 되어야 2로 넘어갈 수 있기 때문에 4명을 붙였으며, 4명은 신속히 1)을 이해해야 한다. – 우선 이해하기 위해서 관련 오픈 소스 자료들을 찾아보고 그 중 가장 우리 주제에 활용하기 좋은 것을 골라 처음부터 끝까지 오픈소스 내용을 최대한 이해한다. 오픈 소스에 대한 이해가 끝나면, 해당 자료를 우리 내용으로 소화하는 작업에 들어간다. ↓

- 우리 주제의 설계 구체화

- 어플리케이션 제작과 연동하여 어떻게 분업할지 정함.

<협업 끌어올리기>

- 분야에 대한 이해도를 신속히 높이기 위해선 소통의 과정이 필수이며, 그렇기 때문에 공부한 내용, 혹은 팀원에게 알리고 싶은 내용 혹은 질문하고 싶은 내용은 되도록 공유할 것

- 화요일 6시 30분에 모여서 회의 진행 또는 정보 공유. [온라인은 시간 관계없이 적극적 활용]

해야할 것 [급선무]

오픈 소스 찾고 공유하여 그중 괜찮은거 선택하기

**10.1 회의 내용 정리**

* 자바로 결정
* 파이썬은 활용할 수 있는 자료가 많음 하지만 파이썬을 새로 공부하는 것이 부담 스러움. 또한 어플과 연동하기 어렵기 때문에 자바로 결정함
* 우리 프로젝트에 딱 알맞은 자바로 구현된 오픈소스가 없음. 하지만 우리 프로젝트의 프로세스별 필요한 자료는 있음. 각각 독립된 자료들을 모아서 활용하여 프로젝트 구현하기로 결정함
* 그래서 자료들을 찾아본 결과

|  |  |  |  |
| --- | --- | --- | --- |
|  | 크롤링 | 형태소 분석 | 워드 클라우드 |
| 오픈소스 |  | <https://github.com/twitter/twitter-korean-text> |  |
| 참고자료 | <https://partnerjun.tistory.com/42?category=693285> [jsoup 사용법이랑, 로그인 필요한 페이지에서도 쓰는법 나와있음]  <https://shlee0882.tistory.com/107>  로그인 정보를 요청,응답시 필요한http프로토콜과 쿠키에 대한 글  <https://github.com/jhy/jsoup> (jsoup관련) | 코모란 (하이퍼링크가 길어서 밑에 첨부) | <https://epicdevs.com/m/5>  <https://github.com/kennycason/kumo> |
| 비고 :  **전체적인 프로세스에 참고될 자료**  <http://hjstudy.blogspot.com/2017/05/java-word-cloud.html>   * 자바로 워드클라우드 구현해보기   (웹페이지 긁어서 string으로 만듦 -> string을 형태소 분석 -> 분석한 자료를 워드 클라우드 자료로 가공 -> 워드클라우드 구현  (완벽한 코드는 아니지만 대강 흐름을 읽을 수 있음)  **워드클라우드에 필요한 자료는 크롤링과 형태소분석이 어느정도 되면 다시 결정하기로 함**  **참고자료**  코모란: <https://shineware.tistory.com/m/entry/KOMORAN-ver-20-beta-%EC%9E%90%EB%B0%94-%ED%95%9C%EA%B5%AD%EC%96%B4-%ED%98%95%ED%83%9C%EC%86%8C-%EB%B6%84%EC%84%9D%EA%B8%B0>  코모란 활용 : <https://github.com/shin285/KOMORAN> | | | |

지금은 전과정을 빠르게 돌려보는데 의의가 있기 때문에

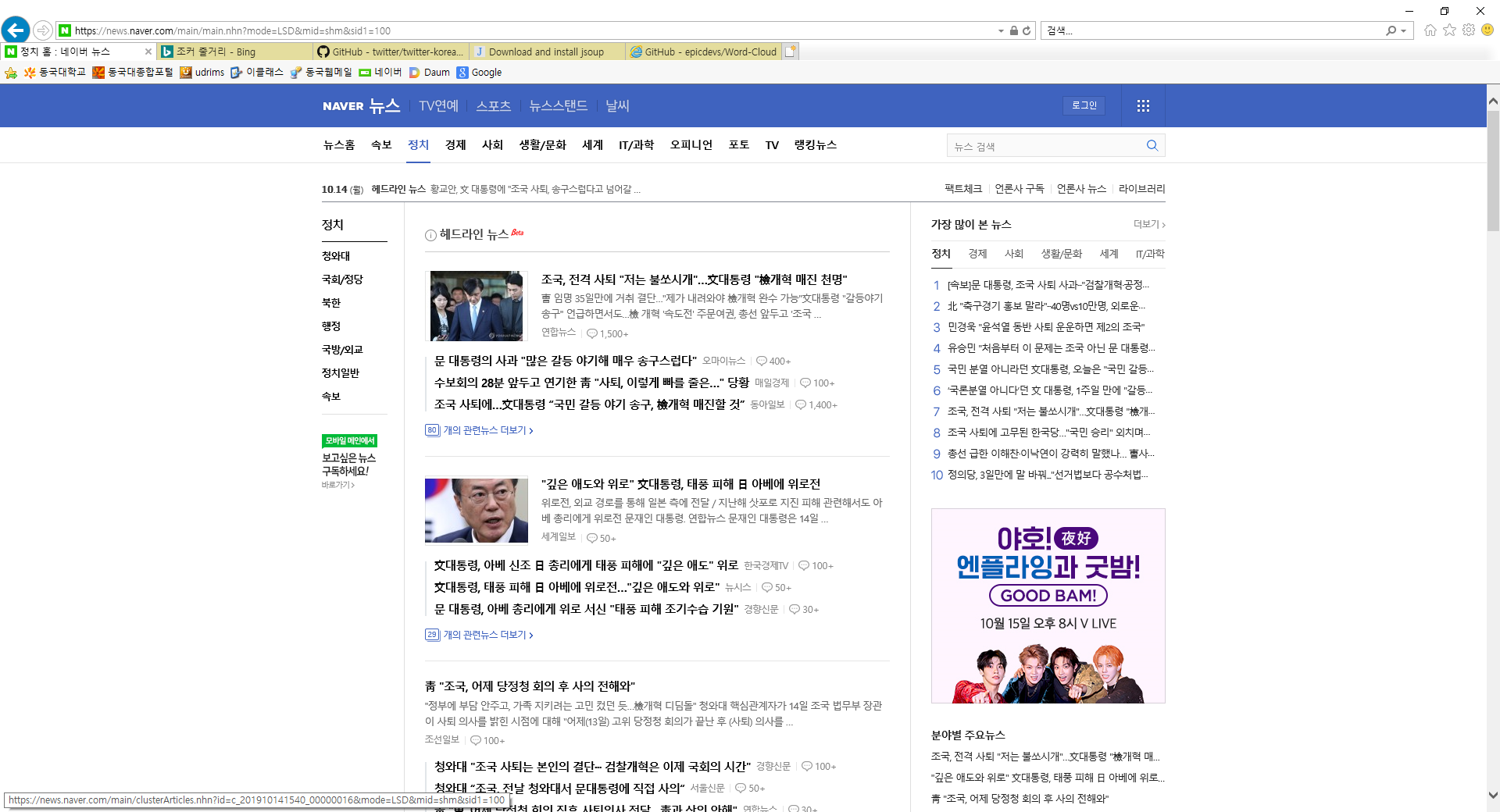
크롤링 할 때, 복잡한 과정은 생략(로그인과 같은 장애물)하고 string우선 뽑고 바로 형태소 분석으로 넘어감. 워드클라우드까지 다하고 나서 크롤링 작업에서 필요한 좀 더 세부화된 내용 이해한다.

10-14 회의록

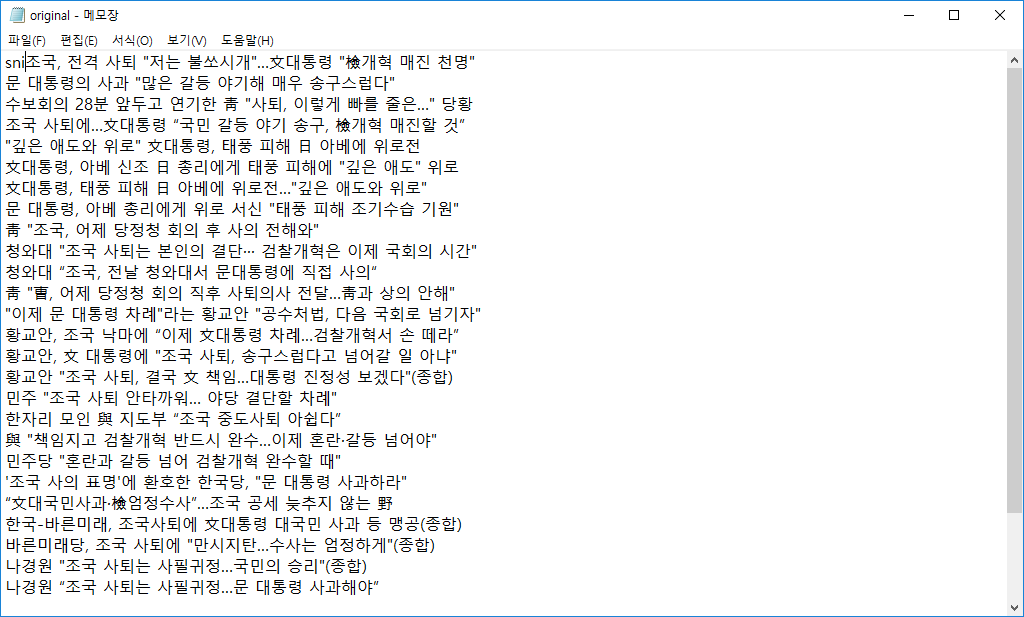
지금까지

프로세스 전체적인 감각을 익히기 위해서 크롤링 형태소 분석 워드클라우드에 사용될 오픈소스를 찾아보고 이들을 연동한 테스트를 해보았다.

네이버 뉴스 기사

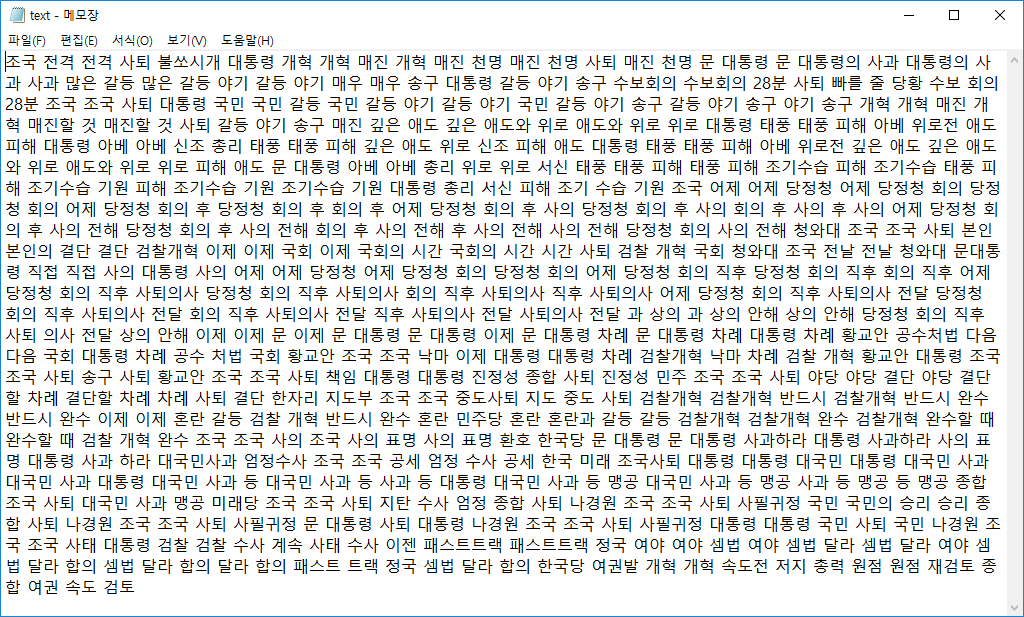


원본



형태소 분석 후 추출

(<https://github.com/twitter/twitter-korean-text> 관련내용)



워드클라우드



기존 예제 프로세스에서 우리가 필요로 하는 부분을 덧붙이고 보완할 계획을 수립함.

1. 크롤링 : 에브리타임 웹사이트를 긁어와야 하는데, 로그인으로 추정되는 문제로 게시글을 읽어오지 못하고 있음. 여기를 뚫을 필요성이 있음.
2. 워드클라우드 : 형태소 분석한 자료를 바탕으로 워드클라우드를 만들어보았지만, 그걸로는 화제를 추측하기 어려웠다. 단어들이 모두 개별적으로 나열되어 있어서 그 단어가 무슨 사안을 담고 있는지 한눈에 파악하기 어려웠음.

그래서

크롤링한 정보에서 단어들을 추출하고 빈도수를 바탕으로 워드 클라우드를 만든다. 워드클라우드로 시각화된 단어들 중에서 관심 있는 대상을 사용자가 선택하고 토론의 방에 들어감. 그 토론의 방에는 에브리타임에서 일정 이상의 관심을 받은 게시글 몇 개가 올라가 있어 토론의 방향을 정해준다. 토론방의 내용도 다시 워드클라우드에 반영이 되게 한다.

(추후 논의 주제 : 일정 시간 지나서 또 다시 크롤링 한 정보는 기존의 정보와 어떻게 합쳐질 것인지. 기준선정.)

1. 워드클라우드를 안드로이드 상에서 우리가 만든다. 링크 기능 걸기 쉽기 때문에.

곁다리 회의 내용..

채팅방 vs 게시판

색깔로 의견 나타내기

**10 21 회의 정리**

지난 주 전까지 진행상황

지금까지 프로젝트를 이해하기 위해 프로세스 구현에 필요한 기술들을 조사하고 이를 바탕으로 매우 간단한 예제를 설정 및 구현을 해보아 프로젝트에 대한 감을 키워보았다. 하지만 아직 크롤링 해온 데이터를 통해서 어떻게 화제를 추출할지에 관한 문제와 안드로이드 스튜디오를 활용해 워드클라우드를 간단히 그려보는 작업은 조금 더 신경쓸 필요가 있었다.

지난 주 사항

사이트에서 글들을 가져와 올바른 한국어로 가공 후, 이를 형태소 분석함으로써 명사를 추출하였고, 빈도수가 높은 단어들의 크기를 높이는 방법은 워드클라우드를 만들기에는 적합하였으나, 화제성이 잘 드러나지 않았다. 지난 주에는 이 부분을 해결하기 위한 아이디어를 생각해 보았다.

화제를 살리는 방법에는 1) 품사별 조합을 통해 짧은 어구를 생성하는 방법과 2)화제성이 강한 단어와 그렇지 않은 단어를 엄격히 분별하는 방법이 있었다. 첫번째 방법은 인공지능 방법이 필요하고 너무 복잡하였기 때문에 두번째 방법을 살리기로 하였다.

두번 째 방법 : 화제성을 살리는 역할을 우리가 아닌 사용자들이 하게하면 가능할 것 같다는 생각이 들었다. 단어의 빈도수를 바탕으로 워드클라우드를 만들어서 보여주고, 사용자들이 워드클라우드를 보고 더 반응하는 단어들의 크기는 키우고 관심을 받지 못한 단어는 시간이 지남에 따라 삭제함으로써 정말 화제성이 강한 단어만 남겨두는 방법이다. 사용자들의 반응은 워드클라우드 상의 각 단어들을 클릭했을 때 나오는 토론의 방에서 사용자들이 게시글을 얼마나 올리는지를 바탕으로 측정한다.

그리고

기술적으로 아직 해결되지 않은 부분은 분업을 통해 담당하기로 하였다.

로그인이 필요한 사이트에서 글을 읽어오는 문제 : 이경현, 오정소

자바와 mySQL을 연동한 DB 구현 : 김규민

mySQL과 안드로이드 스튜디오 연동하여 워드클라우드 만들기 : 유준모 정지원

이번 주

금주에 시험이 몰려 있어 큰 진전은 없었으며 모두 각자 분야를 공부중임. 지난 주 해결하지 못한 부분을 금주 시험이 끝나는대로 최대한 해결하여 예제를 빨리 완성할 계획임. 추후엔 완성된 예제를 바탕으로 우리 주제에 대한 세부적인 설계를 (use case class, class diagram) 세우고 설계에 따라 역할 분배를 다시 할 것임.

**10 28 내용 정리**

5주차 진행사항

에브리타임 웹사이트에서 크롤링을 통해 안드로이드에서 워드클라우드로 간단하게나마 구현해보는게 첫 과제이다. 이를 위해 다음과 같은 일련의 과정들이 해결되어야 했는데 역할 분담을 통해 맡아보았다.

1. 로그인이 필요한 웹페이지 크롤링
2. 크롤링 해온 자료를 데이터베이스에 저장
3. 데이터베이스와 안드로이드 스튜디오 연동
4. 안드로이드 스튜디오 상에서 워드클라우드 그리기

* 로그인

에브리타임은 로그인을 해야 게시글을 가져올 수 있기 때문에, 쿠키와 리퀘스트 헤더를 얻어오는 별도의 작업이 필요했다. 이 둘을 얻어와서 자바를 통해 로그인하는데는 성공은 했지만, 우리가 접근하고자 하는 게시판은 여전히 크롤링이 되지 않았다. 특정 서버가 이 게시판을 통제하고 있다고 추정된다. 이 부분은 좀 보완이 필요하다.

* 크롤링 해온 자료를 데이터베이스에 저장

데이터베이스 툴은 mySQL을 사용했다. 크롤링을 성공했다고 가정하고 데이터베이스에 자료를 넣는 과정을 해결해보았다. 크롤링을 하면 단어와 빈도수가 추출이 되기 때문에, [단어 : 빈도수] 형식의 간단한 테이블을 구상해볼 수 있었고, 이 테이블에 맞게 크롤링한 정보를 가공하여 mySQL에 저장해보았다.

* 데이터베이스와 안드로이드 스튜디오 연동

데이터베이스에 저장된 자료들을 안드로이드 스튜디오 상에서 사용하기 위해선 mySQL과 연동이 되어야 한다. 하지만 보안상의 문제로 스튜디오와 mySQL의 직접 연동이 불가능해 대신 php를 매개로 하는 방법을 시도해보았다. 하지만 Php를 이용하려면 아파치라는 서버에 대한 이해도 필요해서 공부를 더 해야 해결될 것 같다.

* 안드로이드 스튜디오 상에서 워드클라우드 그리기

mySQL에서 자료를 받아왔다고 가정하고 임의의 자료를 갖고 아주 간단하게 인터페이스를 구현해보았다. 아주 간단하게 단어들을 배치해보았고 이 단어들을 눌렀을 때 채팅방으로 이어지게까지 해보았다. 위 과정들이 모두 연결이 되면 DB에 저장된 단어들로 대체될 것이고, 빈도수에 따라 크기가 달라질 것으로 예상하고 있다.

6주차 진행계획

금주에 시험이 모두 끝나기 때문에 이번주 내로 위 과제를 해결하고자 한다. 4과정이 모두 연동이 되면 우리 조 프로젝트를 위한 기본 뼈대가 완성되는 것이다. 이 뼈대 위에서 어떤 서비스를 어떻게 제공할지에 대한 논의를 다시해보고 보다 더 섬세한 작업을 재분담을 통해 해결하고자 한다.

**11/1회의내용**

해결해야할 부분

1. DB & 스튜디오 연동 문제
   1. DB에 저장된 크롤링 자료를 서버에 올리고
   2. 서버에 올라간 내용을 안드로이드 스튜디오에서 가져오기
   3. 가져온 내용을 바탕으로 워드클라우드 그리기
2. 에브리타임 게시판 접근 문제

하지만 에브리타임 게시판 접근 문제는 상당히 시간이 소요될 거 같아서 차차 진행하는 방향으로 결정함. 그러면 이제 연동 문제만 남는데, 연동 문제에 4명 (정지원 제외/ 스튜디오 담당) 이 모두 집중하는 건 비효율적이라고 판단, 다음 진행사항을 결정해서 역할 분배의 필요성을 느낌.

다음 진행 사항을 생각해보기 위해 use case 다이어그램을 바탕으로, 사용자가 우리 어플을 사용할 때의 과정을 상상해보면서 우리가 무엇을 구현해야 하는지 방향을 정립함. 과정은 다음과 같음.

1. 로그인 : 에브리타임 ID,PW을 입력하면 어플에 접속이 가능함
2. 워드클라우드가 바로 보임 : 접속한 시각
3. 하단에 탭을 두어 과거 화제를 알 수 있게 함.

로그인 구현은 에브리타임 로그인 문제와 연동되므로 잠시 미뤄두고 2번과 3번에 집중하기로 함.

2번에서 보여지는 워드클라우드는 어느 시점의 크롤링 된 자료이며, 3번 과거 화제 탭은 어떻게 구현할 것인지에 대해 구체화 작업이 필요하여 회의를 통하여 틀을 잡아봄.

우선 시나리오는 다음과 같음.

메인 화면에 뜨는 워드클라우드는 접속 시각 또는 새로고침 버튼을 눌르면 그때마다 크롤링 된 자료를 바탕으로 한 것임.

하루가 지나면 하루 동안의 크롤링된 정보 상에서 랭킹을 정해 DB에 저장함.

2일차가 되면, 2일차에서도 똑같이 접속되는 시간, 새로고침 되는 시각에 맞춰 워드클라우드를 보여주면서, 다음날이 되면 2일 동안 크롤링 된 자료 상에서 랭킹에 맞춰 잘라서 DB에 저장함.[매일 누적된 자료를 바탕으로 메인에 워드클라우드를 만들어 보여주면, 최근 이슈화되고 있는 화제가 과거 화제의 빈도수 때문에 묻히기 때문에 다음날이 되면 다음날 수집된 자료를 바탕으로 크롤링함. ]

이런 식으로 매일 누적되면, 데이터를 다시 조합하여, 월별 연별로 자료를 묶어서 이를 바탕으로 워드클라우드를 재구성하면 과거 특정 월의 화제, 특정 년도의 화제를 보여줄 수 있음.

이때 토론의 방에서 각 단어별 토론의 정도에 따라 화제성도 달라져야 하는데, 이 빈도수를 100프로 반영해버리면 다른 화제가 압도적으로 묻힐 가능성이 있으므로 가중치를 부여하기로 함.

--

이제 앞으로 무엇을 해야할지 명확해졌음.

기존)

* 로그인문제
* 연동문제

추가)

* **일일 단위 크롤링한 걸 워드클라우드로 구현하기**
* **날이 지날때마다 순위로 잘라 데이터 누적시키기 (과거 탭을 위해)**

경현이와 정소는 <추가>부분에 투입되고, 준모와 규민이는 연동문제를 빠른 시일내에 마치고 <추가>부분에 합류해서 다시 세부 분업에 들어간다.

그동안 지원이는 기초적인 디자인을 해결한다.

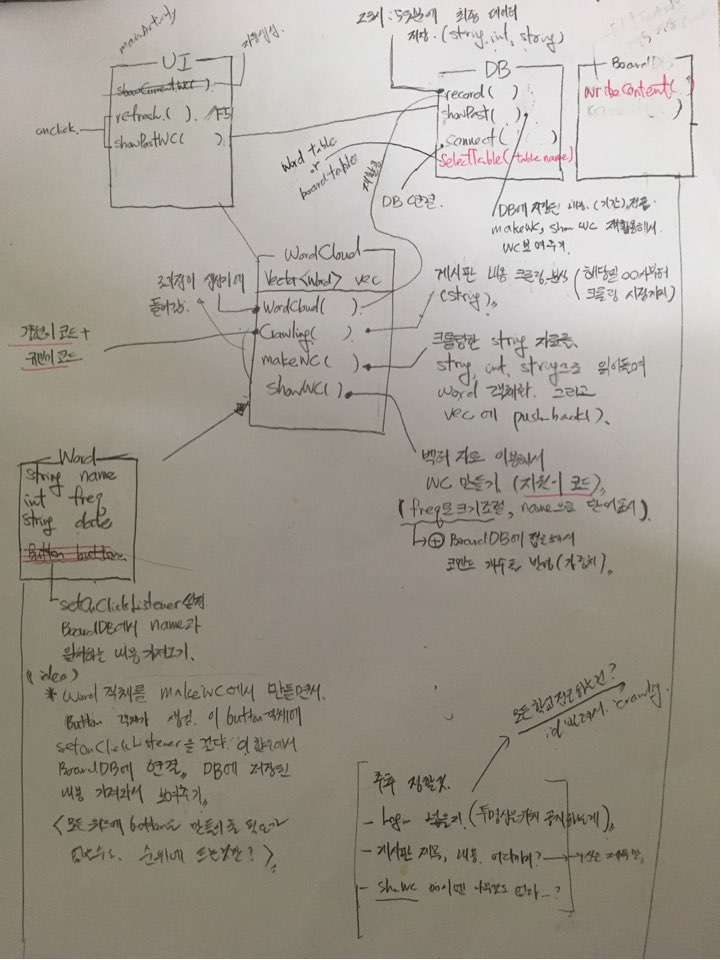
* 빈도수에 따라 글자의 크기가 바뀌고
* 랜덤한 위치에 글자가 뿌려지게 하기

11/5 회의 내용 정리

상황 : 로그인 문제 해결중 / 크롤링 -> 스튜디오 올리기 연동 해결 /

해야 할 것 : 로그인은 계속 경현이가 맡아주고, 스튜디오 상에서 자료들을 활용하는 코드를 구현해야 함. 이를 위해서 사전 설계를 해야함. 클래스 다이어그램 설계 후, 무엇을 구현해야 하는지 확실히 이해한 후, 역할 분담. / 지원이는 계속 디자인 쪽 맡을 것.

11/12 회의 내용 정리



11 13일 회의 내용 정리

경현이 규민이 코드 합치기 : 에브리타임에서 크롤링한 자료 가공해서 안드로이드 스튜디오에 올리는 과정을 모두 스튜디오 툴에서 해결하기.

준모는 클래스 다이어그램을 바탕으로 최대한 코드 짜놓기

지원이는 인터페이스 구현에 집중하기

정소는 로그인에 집중하기