프로젝트 기술요약서

**[프로젝트 개요]**

1. 프로젝트 명 : 프로젝트 참여 플랫폼
2. 개 발 기 간 : 2020.07.30 ~ 2020.08.20 (3주)
3. 투 입 인 원 : 6명
4. 프로젝트 소개: 다양한 직업의 사람들을 필요로 하는 프로젝트(프로젝트 사업을 말한다)를 등록하고 조건에 부합하는 사람들을 연결시켜주는 플랫폼
5. 개발목표 :

1) 로그인/회원가입 및 프로필 구현.

2) 프로젝트를 등록하고 조건에 준하는 사람들이 참여할 수 있게 한다.

3) 채팅을 통해 오너와 멤버간 대화 할 수 있게 한다.

1. 본인역할 :
2. 1) 클래스 다이어그램 구성

2) 프로젝트 리스트 관리 및 댓글&대댓글 관리

3) 로그인 및 회원가입

4) 프로필 관리 (프로젝트 참여 대기 인원을 수락해주는 기능 포함)

**[개발환경 및 사용기술]**

1. 사용(개발) 언어: Java
2. 사용 O/S : Windows
3. 사용 D/B : Oracle
4. 사용(개발) 툴: Eclipse
5. 웹표준 기술 : HTML, CSS, JavaScript, jQuery
6. 프레임워크 : Spring, MyBatis

**[주요기능(구동원리) 및 특징]**

1. 주요특징 : 등록된 프로젝트에 참여를 원하는 사람은 참여버튼을 클릭, 프로젝트 오너가 수락해야 참여가 확정된다.
2. 주요기능 : 프로젝트 등록 기능, 댓글과 답글 기능, 참여 기능 및 수락 기능, 채팅 기능, 해시태그를 통한 검색 기능
3. 기타사항 : 프로젝트 정렬 기능, 로그인/회원가입 기능 및 권한부여 기능

**[문제해결사례]**

>> 기술부문 : 검색 키워드를 통해 검색을 했을 때 자동으로 최신 순 정렬이 되지 않았던 문제가 있었다. 페이지 처리와 검색 키워드를 통한 검색 기능을 하나의 SQL문으로 작성 해야 하다 보니 어려움이 있었다. 일반 자바에서 처리한다고 생각해보니 if문이 필요했고, 구글을 통해 xml에서 if문을 사용할 수 있다는 것을 알게 되었다. SQL문은 길어 졌으나 검색을 했을 때 자동으로 최신 순 정렬이 되고, 페이지 처리도 되어 있는 결과를 얻어낼 수 있었다.

기술부문 : 참여 버튼을 누르면 프로젝트를 관리하는 오너 프로필에서 참여 버튼을 누른 참여자ID와 참여 프로젝트 명을 함께 띄어주려고 SQL문을 작성했으나 xml에서 두 정보를 담을 하나의VO가 없었다. 클래스 다이어그램에서의 정보를 유지하려고 다양한 시도를 해보았지만 결국 참여자 ID가 있는 PartyVO에 프로젝트 정보를 넣어서 성공시킬 수 밖에 없었다.

>> 비기술부문 : 클래스 다이어그램 구성 시 클래스 구성부터 클래스 사이의 관계를 이해하기 힘들어 시작 할 때 막막했던 문제가 있었다. 선생님께 질문을 하여 힌트를 얻고, 구글을 통해 여러 예시를 찾아보며 클래스 사이의 관계 및 적절한 표현을 알 수 있었다. 그 결과로 우리 조의 클래스 관계에서 프로젝트 게시물과 사용자 간의 관계가 다 대 다 관계 임을 인지할 수 있었고, 그 둘의 클래스 사이에 매핑을 해주는 연결 클래스를 만듦으로써 클래스 다이어그램의 초석을 다질 수 있었다. 그것을 시작으로 순조롭게 진행하여 우리만의 클래스 다이어그램을 구성할 수 있었다.

**[특징 및 주요기능등]**

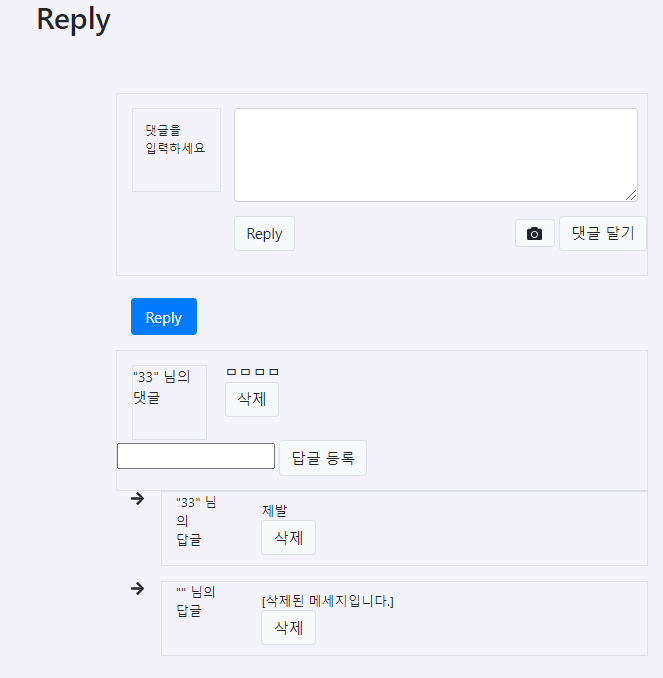
▶ 특징 : KOMORAN을 이용하여 문장에 포함하는 단어들을 추출하여 해시태그 테이블에 저장하였고, 검색어가 제목, 내용, 작성자를 통틀어서 그 안에 포함이 된다면 관련 프로젝트가 나오는 기능을 구성하였다.

▶ 주요기능 : VO에서 제목+내용+작성자를 연결시켜 연결된 내용 가져옴.

Service에서 KOMORAN을 호출하여 연결된 내용을 분리.

분리된 내용을 매퍼를 통해 해시태그 테이블에 저장.

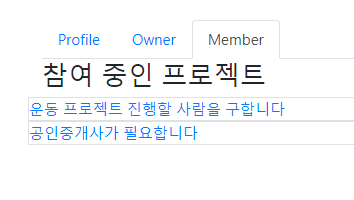
<댓글과 답글>



댓글 입력 부분과 답글을 입력하는 부분입니다.

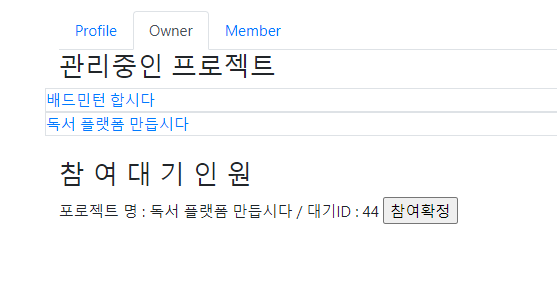
아쉬운 점 : 답글을 등록하기를 누르면 입력란이 뜨게 하고 싶었으나 데이터를 가져오고, 그것을 반복문으로 표현하는데 있어서 어려움을 겪었습니다. 결국, 답글 입력란을 고정적으로 표현 할 수 밖에 없었습니다.

<프로필에서 참여 중인 프로젝트를 표현>



아이디를 통해 참여 중인 프로젝트를 추출하였습니다.

<관리 중인 프로젝트를 표현 & 참여하기를 누른 회원 표현>



select p.\*,tp.\* from tbl\_map\_proj\_party m,tbl\_project p,tbl\_party tp

where m.proj\_id = p.hier\_id

and m.party\_id = tp.party\_id

and p.hier\_id in (select pp.hier\_id from tbl\_project pp, tbl\_map\_proj\_party mm, tbl\_party tp

where pp.hier\_id = mm.proj\_id

and tp.party\_id = mm.party\_id

and mm.party\_type = 'O'

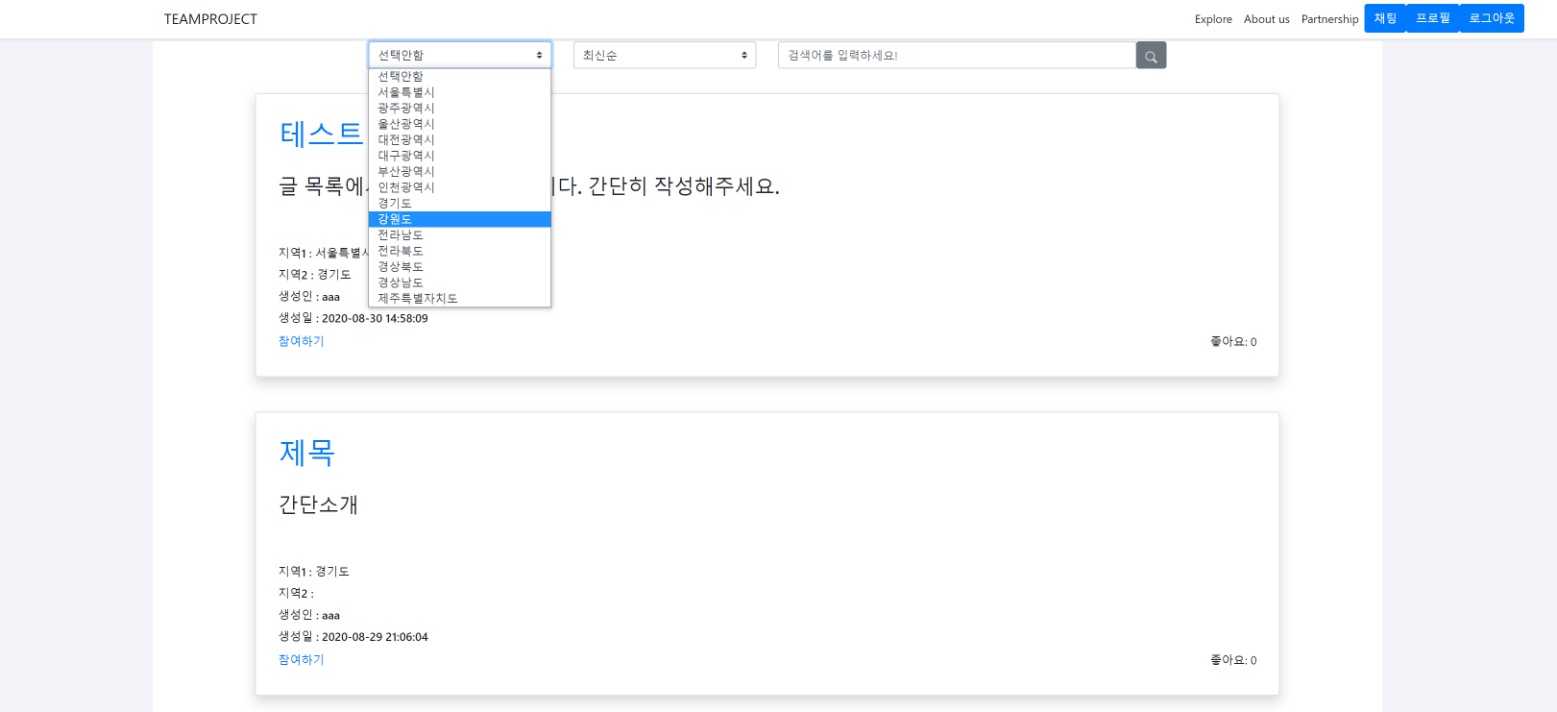
and mm.party\_id = #{partyId})

and m.join\_state = 'wait'

and p.obj\_type='P'

프로젝트 사업을 생성한 프로젝트 오너의 프로필에는 관리 중인 프로젝트에 참여된 프로젝트를 표현 할 수 있습니다. 또한, 참여버튼을 누른 인원이 프로젝트 이름과 함께 표시되게 하였습니다.

<프로젝트 사업 리스트>



올라온 프로젝트 리스트를 볼 수 있습니다. 지역별로 올라온 프로젝트를 검색할 수 있고 최신 순, 좋아요 순으로 정렬 할 수 있는 기능이 있습니다. 또한, 키워드 검색을 통하여 원하는 내용이 포함된 프로젝트 사업을 추출 할 수 있습니다.

아쉬운 점은 정렬하는 부분에서 최신 순, 좋아요 순을 하나의 sql문에 담고 싶었지만, 검색어 부분, 지역 부분, 페이징 처리까지 신경 쓰다 보니 하나의 sql문에 담지 못했습니다. 결국, order by문을 변경하여 2개의 sql문으로 나눌 수 밖에 없었습니다.

select m.\* from (

select ROWNUM rn, p.\*, u.\* from tbl\_project p, tbl\_party u

where hier\_id in(

select hier\_id from tbl\_map\_proj\_tag mt, tbl\_hash\_tag ht, tbl\_project tp

where mt.tag\_id = ht.tag\_id

and tp.hier\_id = mt.proj\_id

and ht.tag\_name = #{searchTag}

and u.party\_id = p.writer\_id

)

<if test=*"searchTag.isEmpty()"*>

or hier\_id in(

select distinct hier\_id from tbl\_map\_proj\_tag mt, tbl\_hash\_tag ht, tbl\_project tp

where mt.tag\_id = ht.tag\_id

and tp.hier\_id = mt.proj\_id

and ht.tag\_name = #{area}

and u.party\_id = p.writer\_id)

</if>

and p.obj\_type = 'P'

order by p.hier\_id desc, creation desc) m

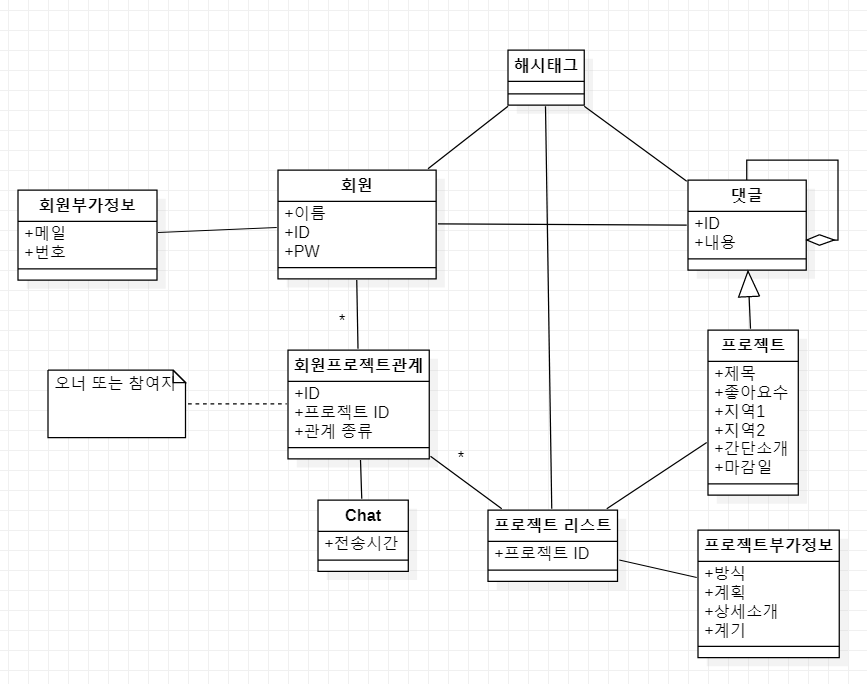
<![CDATA[

where rownum <= #{cri.pageNum} \* #{cri.amount}

and rn > (#{cri.pageNum}-1) \* #{cri.amount}

]]>

<클래스 다이어그램>



깃허브 주소

https://github.com/jeongyeongtaek/Myproject.git

>> 채팅 사용시 이용자 PC의 IP를 chat/list.jsp에서 입력해줘야 이용가능합니다.