

제 10 회 소프트웨어공학과 경진대회

작품 제안서

SM 사업 시스템

상원이를 기억하조

2018 년 3 월

서론

성공회대학교 소프트웨어공학과 학생회 사업에서 출발한 멘토 - 멘티 시스템, 즉 SM 사업은 시작한지 3년째인 지금 소프트웨어공학과와 대표 사업이 되었다.

매년 신입생 및 복학생들은 학과 생활 적응과 학업의 어려움을 겪고 있다. 이에 소프트웨어공학과 학생회는 학과 차원에서 선 - 후배간 멘토링 사업을 통해 선배는 자신의 지식과 노하우를 후배에게 알려주면서 다시한번 복습을 하게 되고, 후배들은 이러한 도움을 통해 빠르게 학과 생활에 적응해 간다. 학과에서는 매번 지원금을 통해 학생들이 더욱 적극적으로 참여하도록 노력하고 있다.

현재까지 SM 사업은 학생회의 노력을 통해 이루어져왔다. 매번 보고서를 받고, 직접 신청을 받는 형식을 해왔다.

본 프로젝트에서는 SM 사업 시스템 구축을 통해 SM 사업을 전산화 한다. 이에 따라 이전 자료를 찾아볼 수 있게 되고, 관리자는 간편하게 SM 사업을 관리할 수 있게 된다. 멘토, 멘티들은 간편하게 웹에서 보고서를 작성하는 것으로 활동을 할 수 있게 된다.

주요어:SM 사업 시스템, Spring-boot, JSP, Mybatis, JPA, Redis, Spring-Security, AWS

목차

1. 개요	1
1.1 주제 / 작품 명	1
1.2 팀원 소개 및 담당 내용	1
1.3 개발 동기	1
1.4 개발 목적	2
1.5 개발 목표	2
1.6 창의성/우수성	2
1.7 활용성/사업성	3
2. 개발	3
2.1 개발 일정	3
2.2 개발 내용	3

3. 구성	4
3.1 전체 시스템 구성	4
3.2 웹 서비스	5
3.3 서버	23
4. 참고 문헌	27

1. 개요

1.1 주제 / 작품 명

주제: 소프트웨어공학과 멘토링 사업 시스템

작품 명: SM 사업 시스템

1.2 팀원 소개 및 담당 내용

- 소프트웨어공학과 201232016 배다슬 / PM, 문서 작성, 백 엔드 서버 개발, DB 설계, 배포
- 소프트웨어공학과 201232003 국중서 / 사용자 인터페이스, 프론트 엔드 개발
- 소프트웨어공학과 201532004 김보미 / 백 엔드 서버 개발, DB 설계
- 소프트웨어공학과 201532016 박소희 / 사용자 인터페이스, 프론트 엔드 개발, 문서 작성, ppt 작성

1.3 개발 동기

- 현재까지 SM 사업은 학생회의 노력으로 이루어지고 있다. 개별적으로 신청을 받고, 보고서를 받는다. 모든 것이 사람손에 의해 이루어지고 있다.

- 이에 따라 소프트웨어공학과와 대표 사업인 SM 사업의 활성화 및 편의성을 위해 전산화 시스템이 필요하다.

- 전산화를 통해 이전 자료를 찾아 볼 수 있고, 사업의 관리도 용이해 질 수 있으며, 사용자들은 간편하게 서비스를 이용함으로써 SM 사업에 참여할 수 있다.

1.4 개발 목적

- SM 사업 시스템 개발

1.5 개발 목표

- Spring framework으로 웹 서비스 개발
- Mybatis를 사용해 MySql DB 사용
- JSP를 이용해 프론트 엔드 구성
- AWS의 서비스를 통해 배포

1.6 창의성 / 우수성

- 부트 스트랩을 사용해 반응형 웹으로 개발했기 때문에 별도의 앱 없이 모바일 환경에서도 사용이 가능하다.

1.7 활용성 / 사업성

- AWS의 제품들을 사용해 배포하기 때문에 유지 보수가 상대적으로 간단하다.

2. 개발

2.1 개발 일정

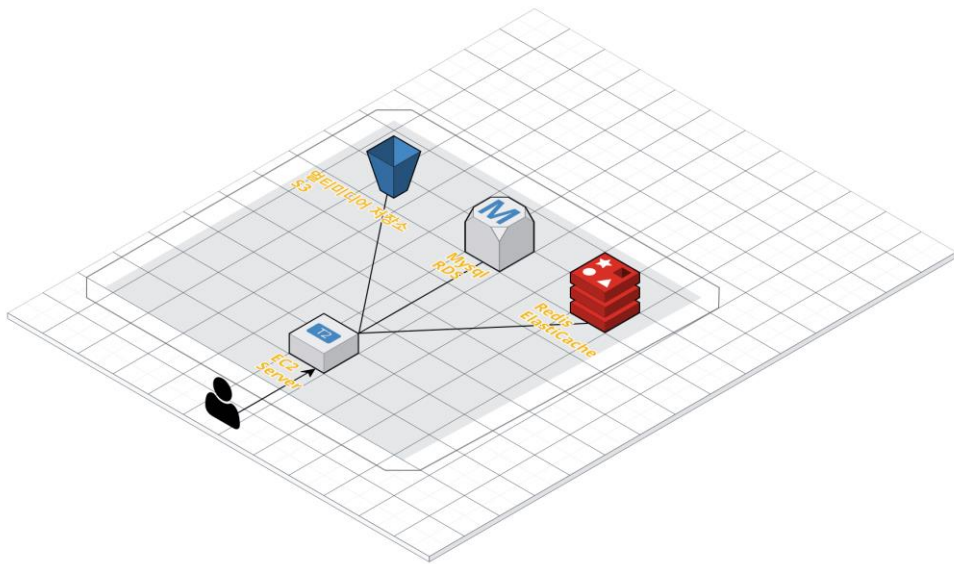
- 2017.08.28~2017.09.01 : 팀 구성 및 목표 설정
- 2017.09.02~2017.09.29 : 요구사항 분석 및 사용자 인터페이스 구성
- 2017.09.30~2017.10.13 : DB 설계 및 프론트 엔드 개발
- 2017.10.14~2017.12.05 : Spring framework 웹 서버 개발 및 배포
- 2018.02.28~2018.03.15 : Redis 연동

2.2 개발 내용

- Spring Framework으로 웹 서버 개발
- My SQL로 웹 서버 DB 구축
- Mybatis 사용
- JPS로 프론트 엔드 구성
- AWS 제품을 사용한 배포
- Redis를 이용한 세션 관리

3. 구성

3.1 전체 시스템 구성

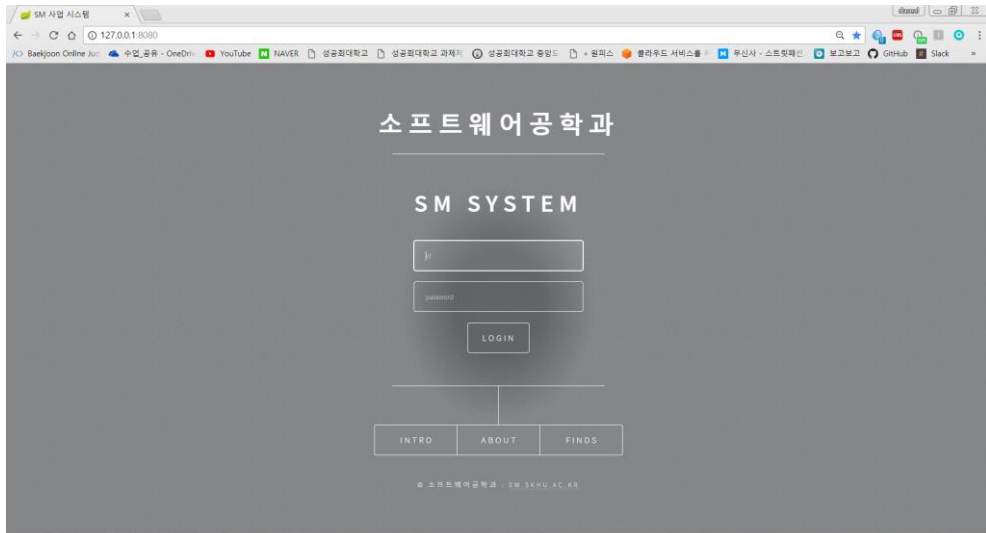


[그림 III-1] 전체 네트워크 구성도

모든 서비스는 AWS의 제품들을 통해서 제공된다.

Spring boot 웹 애플리케이션은 EC2를 통해 배포된다. MySQL은 AWS RDS를 통해 서비스 되고, 멀티미디어 저장을 위해 S3 저장소를 사용한다. Redis을 사용하기 위해 ElastiCache를 사용한다.

3.2 웹 서비스

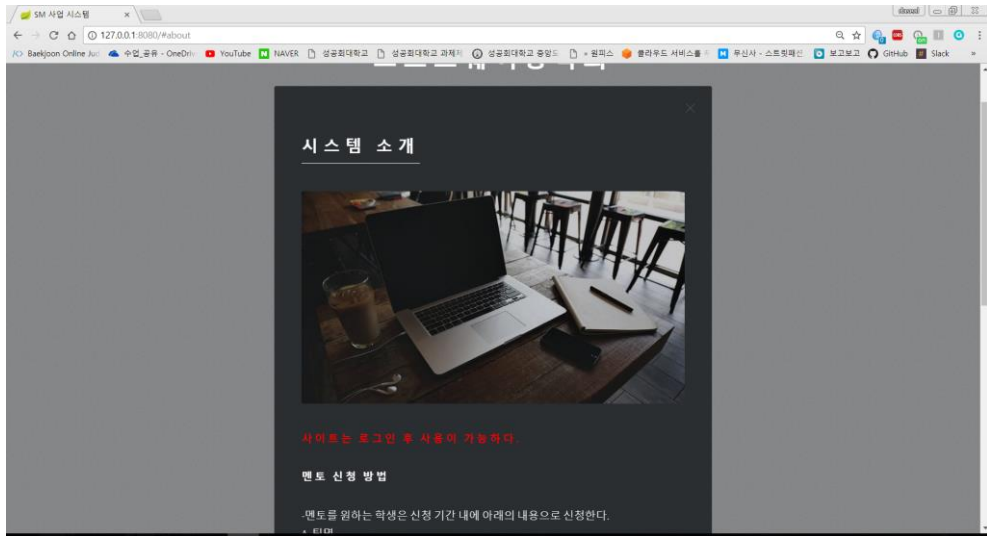


[그림 III-2] 메인 페이지

웹 서비스에 처음 접속하면 나타나는 메인 페이지 이다.

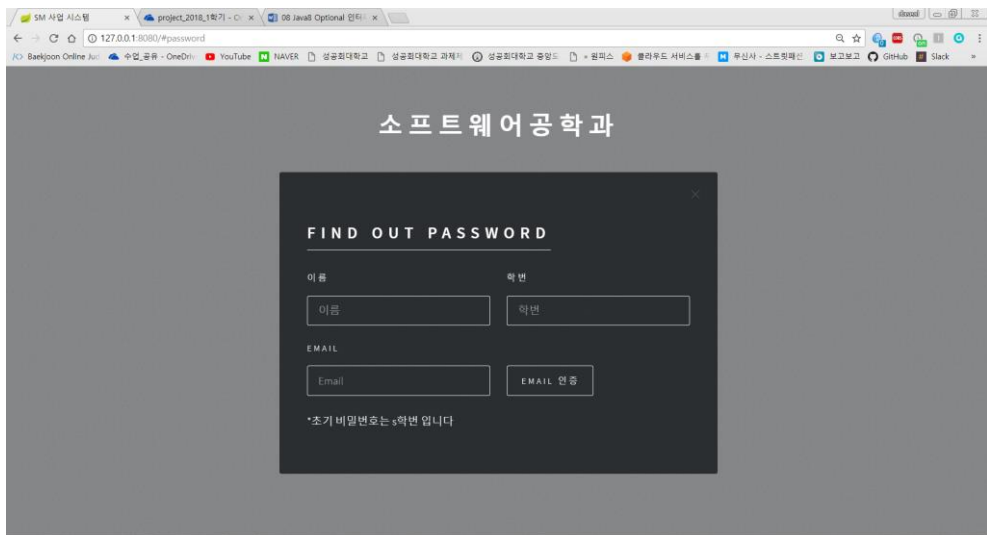


[그림 III-3] Intro 페이지



[그림 III-4] 시스템 소개

로그인을 하지 않아도 Intro, 시스템 소개 페이지를 열람 할 수 있다.



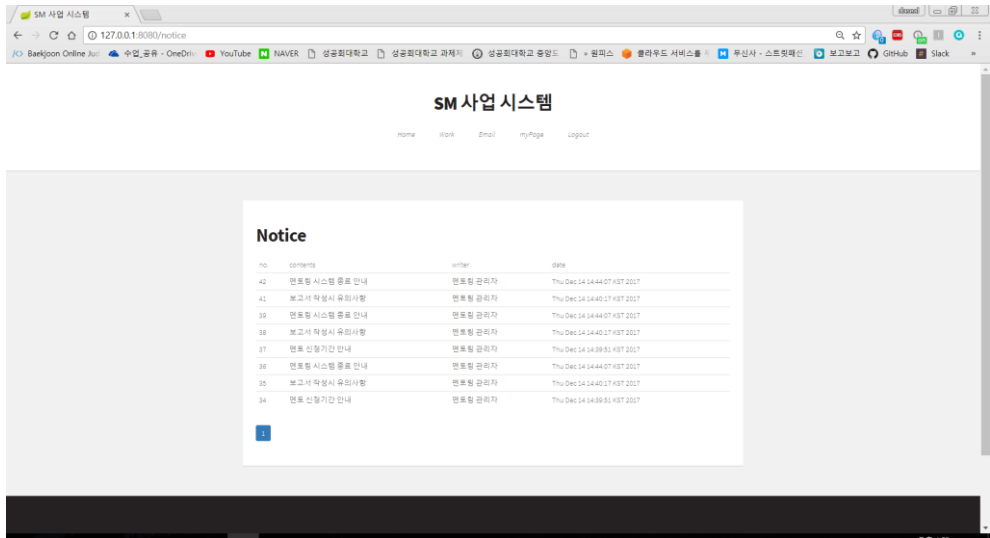
[그림 III-5] 비밀번호 초기화



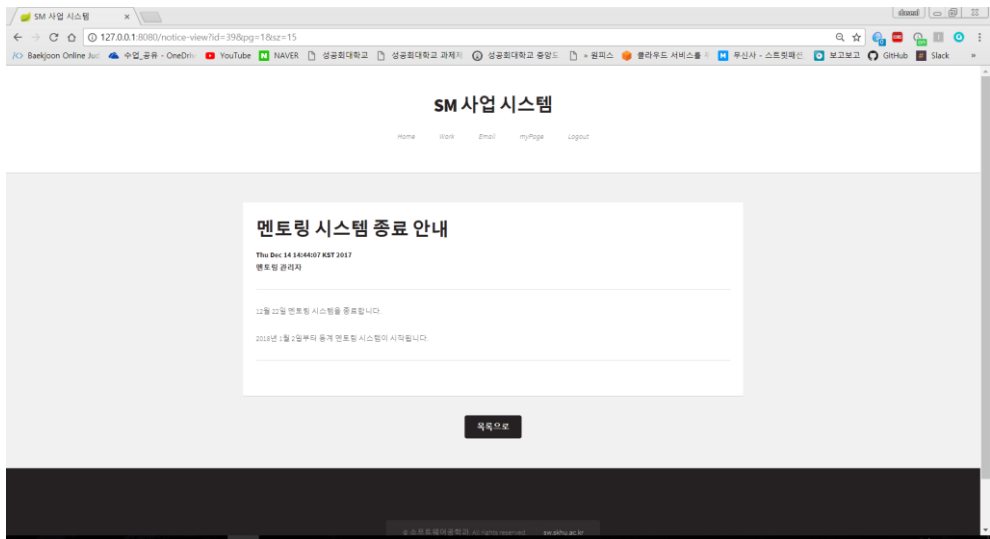
[그림 Ⅲ-6] 중복 로그인을 시도할 경우



[그림 Ⅲ-7] 로그인 정보가 잘못 된 경우

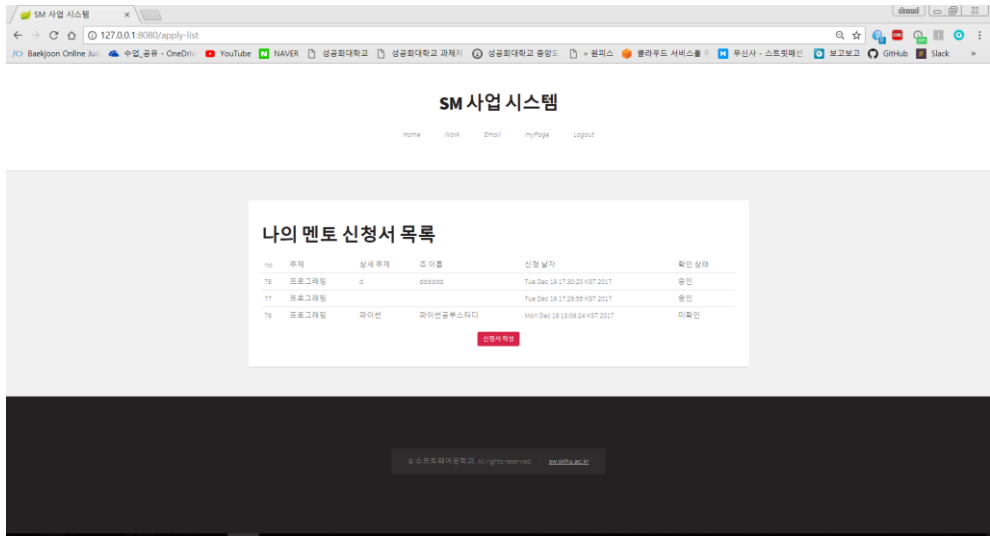


[그림 III-8] 공지사항



[그림 III-9] 공지사항 세부

공지사항은 오직 관리자만 글쓰기 및 수정, 삭제가 가능하다.



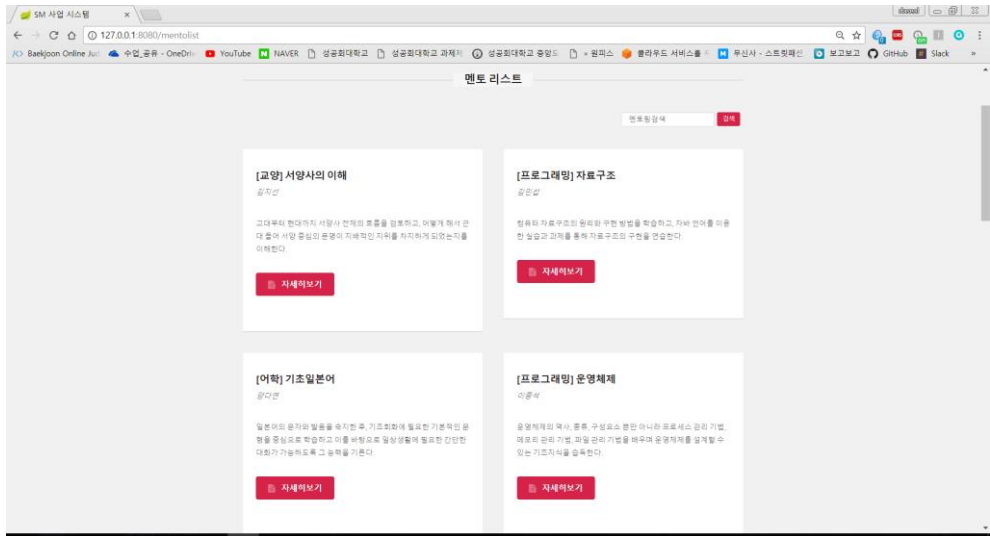
[그림 III-10] 멘토 신청

자신이 멘토로 신청하고 싶을 때 사용하는 메뉴이다.



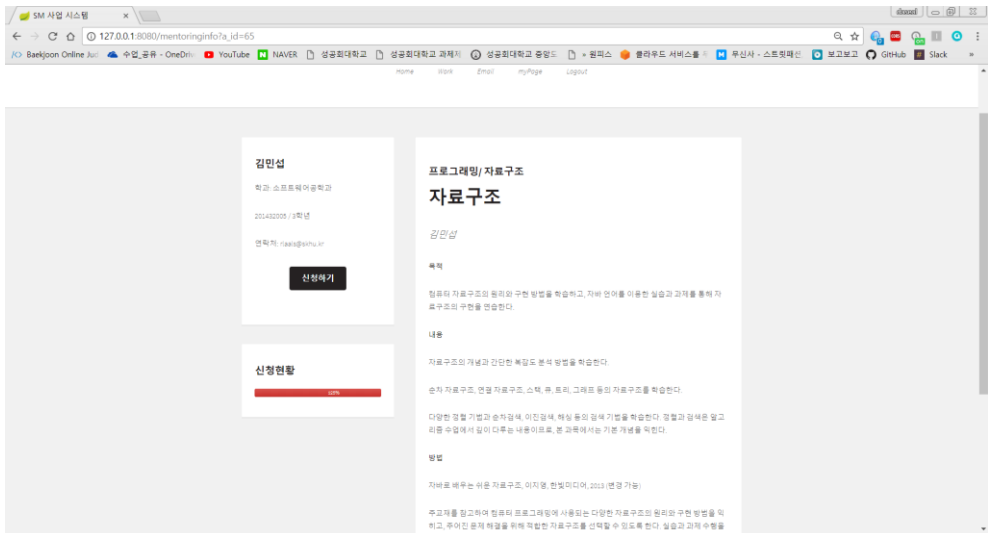
[그림 III-11] 멘토 신청서 작성

멘토 신청서를 작성하고, 관리자는 이 신청서를 토대로 멘토로 승인 여부를 결정한다. 업로드 파일은 S3에 저장된다.



[그림 III-12] 멘토링 리스트

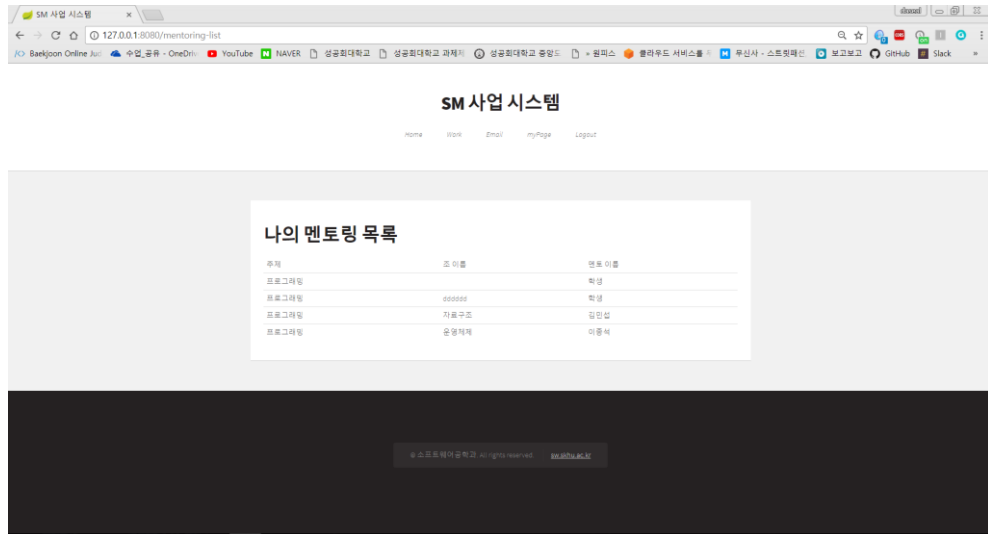
멘토 신청서가 승인 된 후 자동으로 멘토링 리스트에 등록이 된다. 해당 페이지에서는 현재 모집중인 모든 멘토링 리스트를 볼 수 있고, 신청할 수 있다.



[그림 III-13] 멘토링 세부 정보

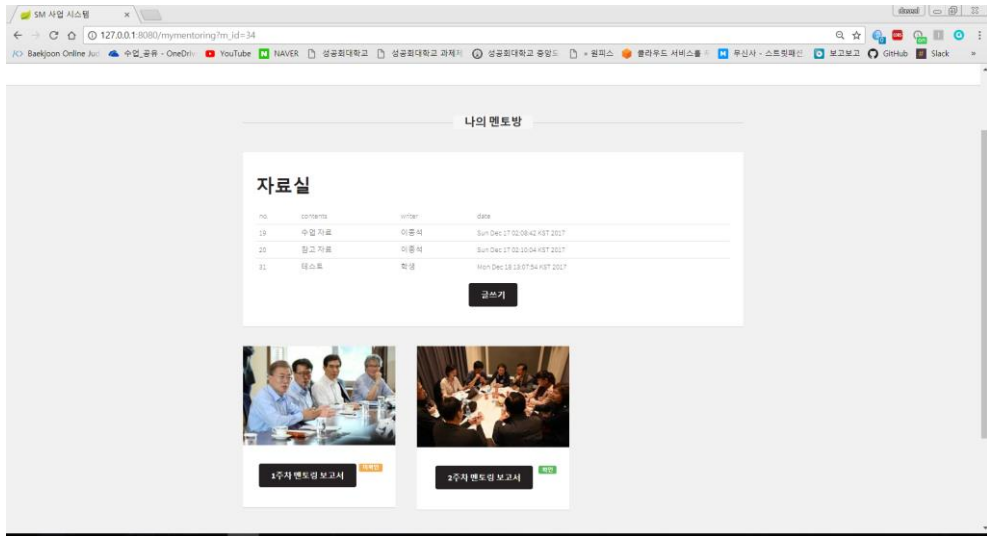
멘토링 세부 정보를 볼 수 있고, 현재 신청 현황, 멘토의 연락처 정보를 확인

해 볼 수 있다.



[그림 III-14] 나의 멘토링 목록

내가 신청한 멘토링 목록이다. 멘토인 경우 내 멘토링 목록이 같이 보이고, 멘티인 경우, 내가 신청한 멘토링 목록이 표시된다.



[그림 III-15] 나의 멘토방

나의 멘토링 목록 중 하나를 선택하면 표시되는 화면이다. 이곳에서 멘토링 활동에 대한 보고서를 작성하거나 볼 수 있고, 자료실을 이용해 자료를 주고 받을 수 있다.



[그림 III-16] 멘토링 보고서

멘토가 작성한 멘토링 보고서 이다. 관리자가 평가한 항목이 상단 우측에 표시된다. 표시 등급은 미확인, 반려, 승인 3단계가 있다. 반려 일 경우 반려 메시지를 확인하고 보고서를 수정 한 다음 다시 제출하면 된다.

SM 사업 시스템

127.0.0.1:8080/report?m_id=47

Backjoon Online Junior 수업,공유 - OneDrive YouTube NAVER 성공회대학교 성공회대학교 과제 성공회대학교 중량도 원피스 온라인 서비스들 우신사 - 스트릿팩트 보고보고 GitHub Slack

멘토 도우미 활동 보고서

멘토 이름 (학생ID)

멘티 이름

주차

활동 날짜

연도 월 일

활동 시간

활동 내용

File Edit View Format

This domain is not registered with TinyMCE Cloud. [Click a free link](#) to discover our premium cloud services and pro support.

활동 소감

[그림 III-17] 멘토 활동 보고서

멘토가 작성하는 멘토 활동 보고서 페이지이다.

SM 사업 시스템

home work email myPage logout

Email

제목

받는 사람

내용

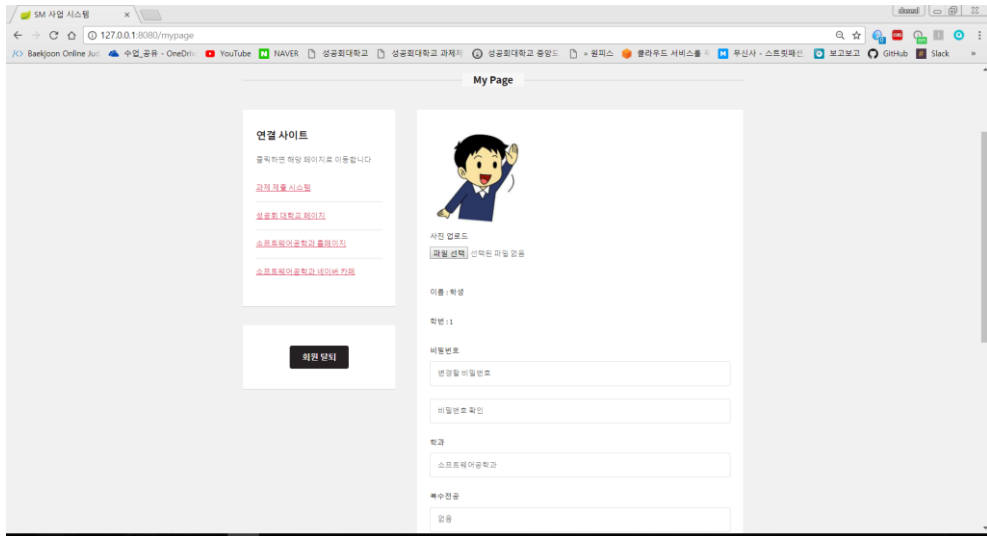
파일 업로드

파일 선택 선택한 파일 업로드

Send Message Reset

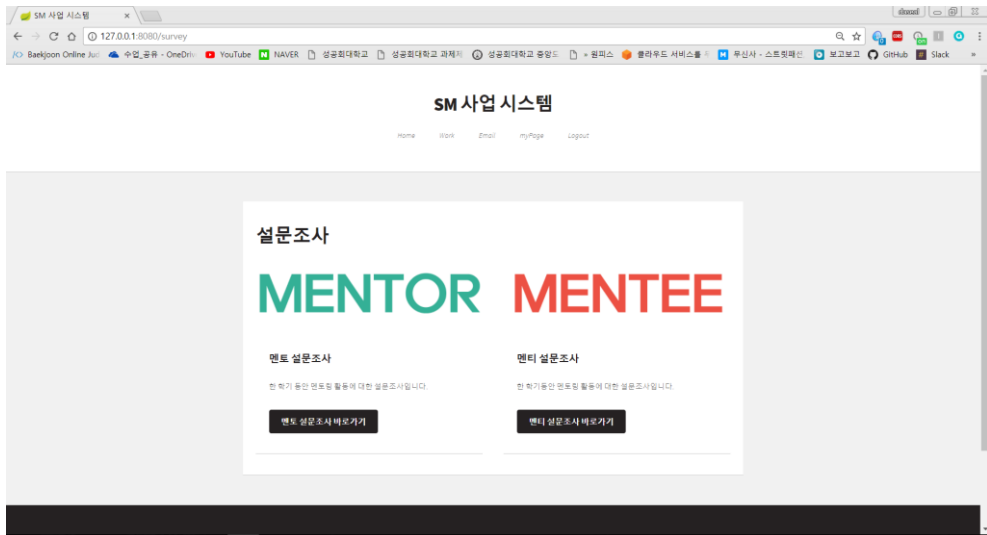
[그림 III-18] 이메일 시스템

이메일 시스템을 이용해 메일을 보낼 수 있다.



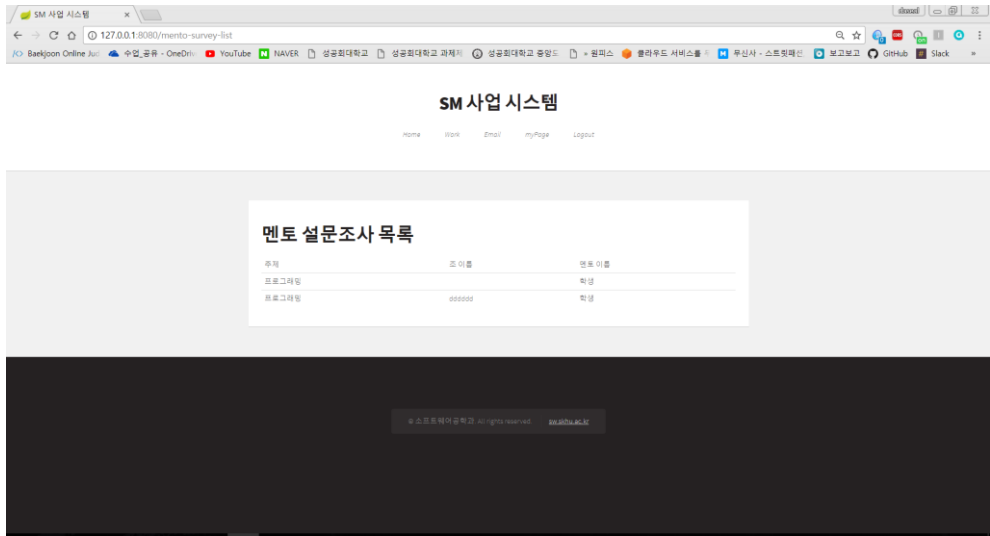
[그림 III-19] 마이 페이지

마이 페이지에서 내 정보를 수정 할 수 있다.



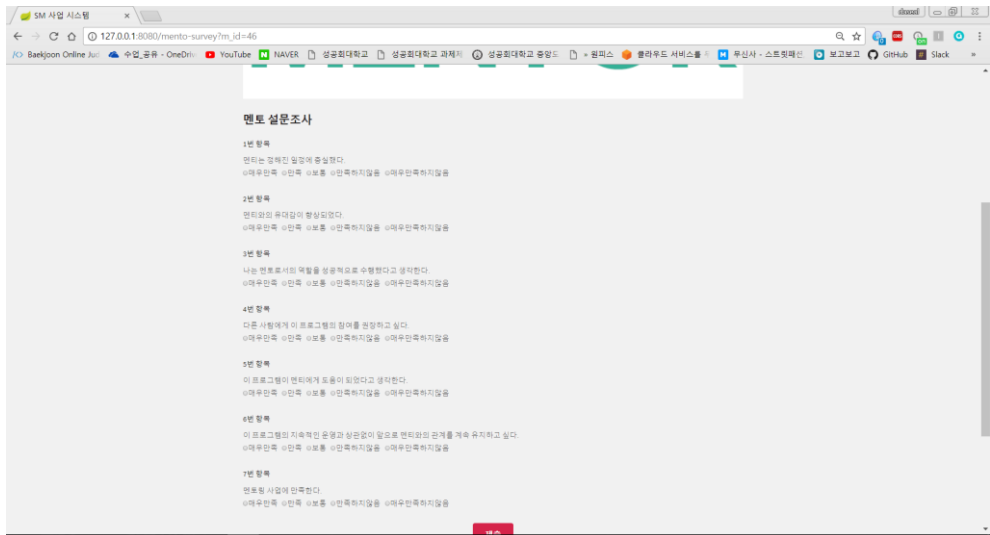
[그림 III-20] 설문 조사

멘토링 활동이 종료 된 후 설문조사를 이용할 수 있다. 멘토는 멘토 설문조사, 멘티는 멘티 설문조사를 이용한다.



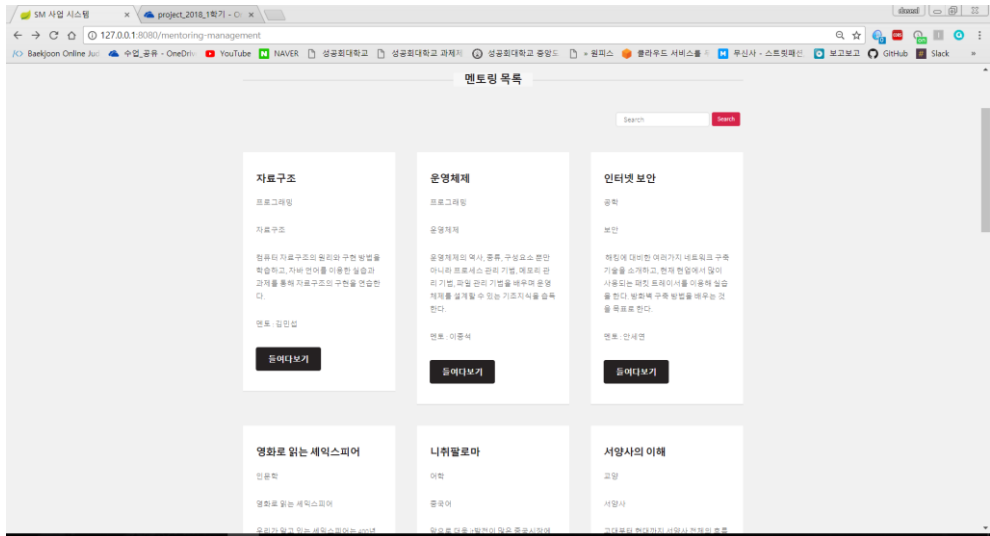
[그림 III-21] 멘토 설문조사 목록

내가 속한 멘토링 마다 설문조사를 진행한다.



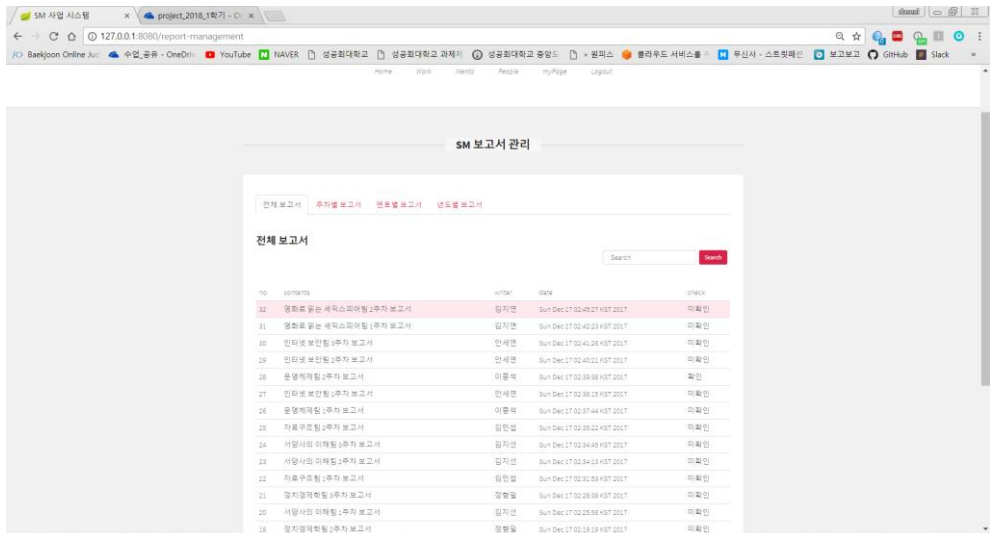
[그림 III-22] 설문 조사

설문 조사 페이지이다. 모든 설문조사는 익명으로 저장되며, 결과는 관리자만 볼 수 있다.



[그림 III-23] 멘토링 관리

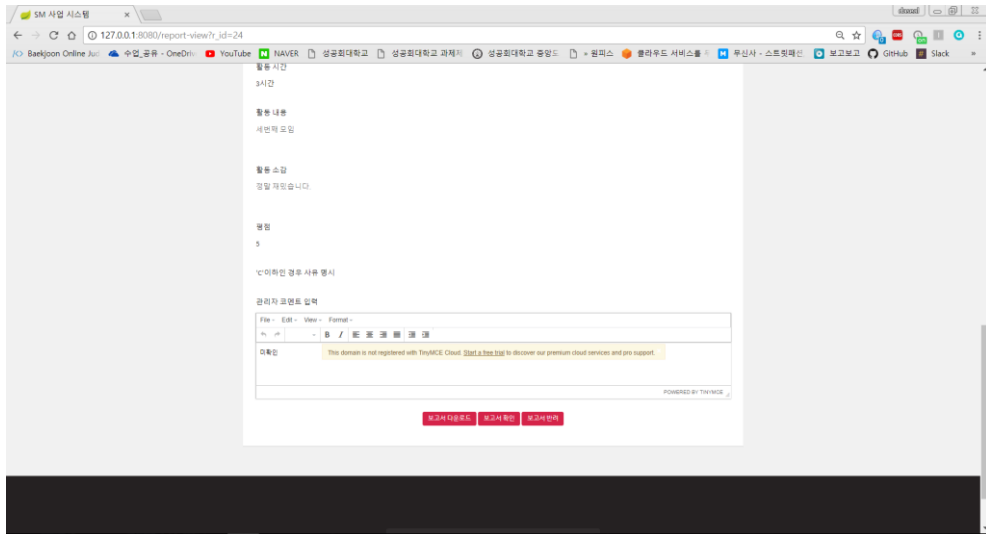
관리자는 모든 멘토링을 관리한다. 현재 진행중인 모든 멘토링 활동을 볼 수 있다.



[그림 III-24] 보고서 관리

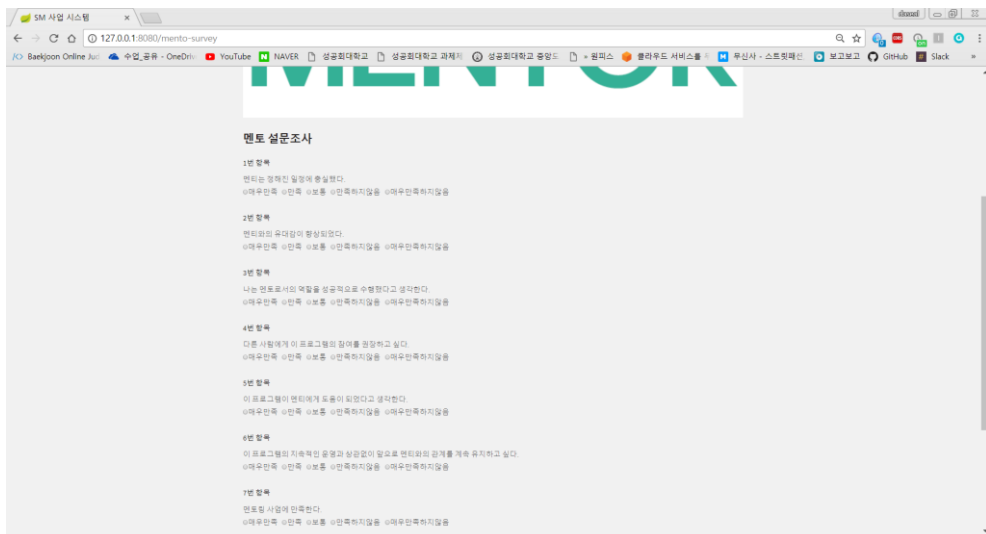
멘토들이 작성한 보고서를 이곳에서 관리한다. 년도별, 주차별, 이름별로 보고

서를 검색할 수 있다.



[그림 III-25] 보고서 확인/반려

관리자는 보고서를 확인하고 보고서를 확인하거나, 반려 메시지와 함께 반려할 수 있다.



[그림 III-26] 설문조사 항목 수정

관리자는 설문조사 항목을 수정할 수 있다.

설문조사 항목	응답	비율
멘토는 일찍인 일정에 충실했다	1	20.0%
재우만족	3	60.0%
인족	1	20.0%
보통	0	0.0%
안락하지않음	0	0.0%
재우만족하지않음	0	0.0%

설문조사 항목	응답	비율
멘토와 유대감이 형성되었다	1	20.0%
재우만족	3	60.0%
인족	1	20.0%
보통	0	0.0%
안락하지않음	0	0.0%
재우만족하지않음	0	0.0%

[그림 III-27] 설문조사 결과

멘토 / 멘티 설문조사 결과를 볼 수 있다. 모든 설문조사는 익명으로 진행되기 때문에 누가 무슨 항목에 투표했는지 알 수 없다.

SM 사업 시스템

home work mentor people myPage Logout

멘토 신청자 리스트

박종길의 멘토신청서

함영희의 멘토신청서

멘토 신청서

스타디 주재 : 프로그래밍 / 파이썬

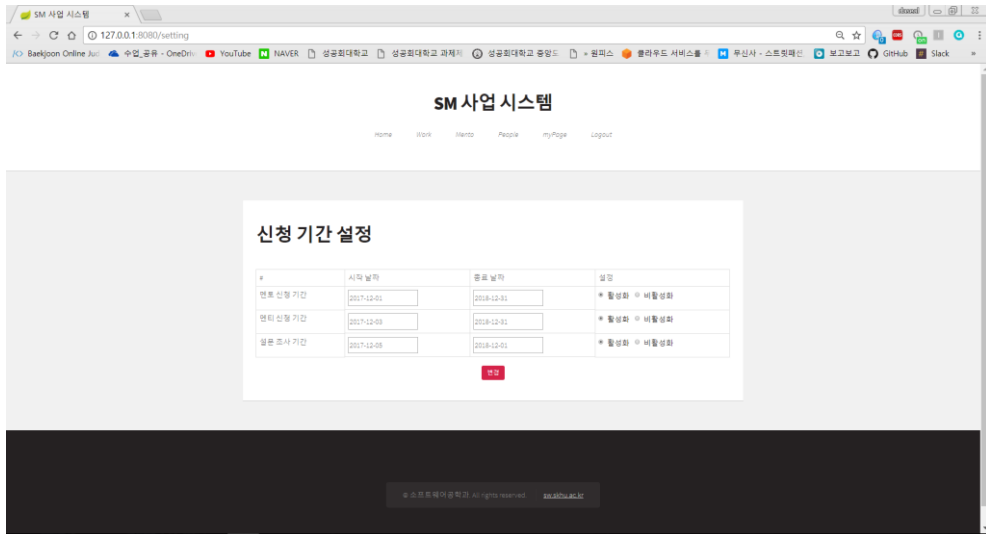
이름 : 학생

학번 : 4

학번 : 1

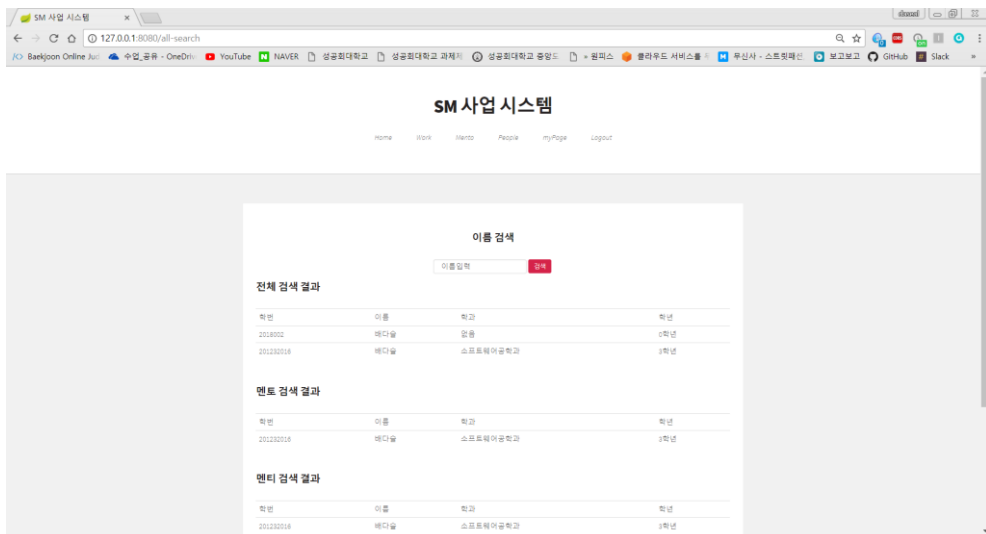
[그림 III-28] 멘토 신청서 확인

멘토 신청자들의 신청서를 확인하고 그들을 멘토로 승인하거나 반려할 수 있다.



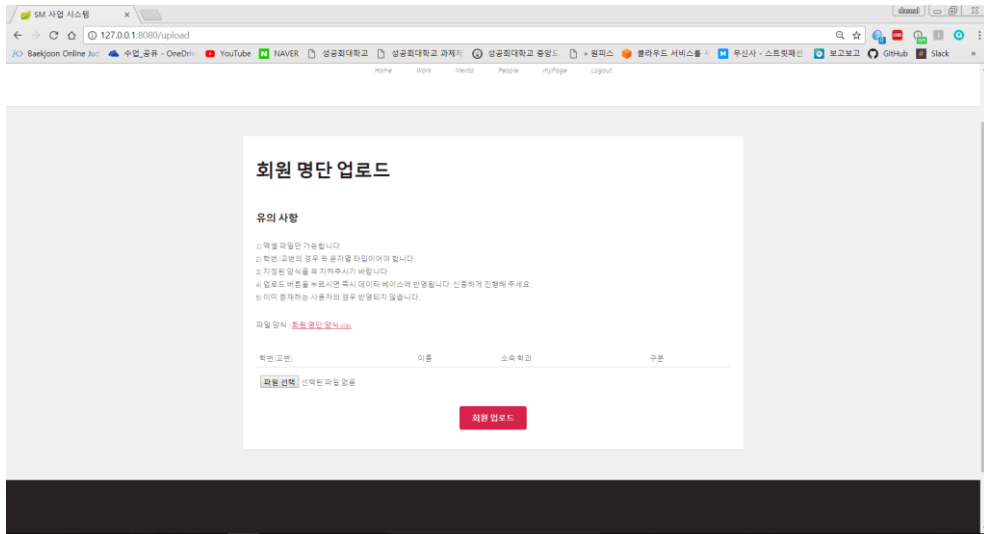
[그림 III-29] 신청 기간 설정

멘토, 멘티, 설문조사 기간을 설정할 수 있다. 기간이 설정되면 해당 기간엔 서비스를 이용할 수 없다.



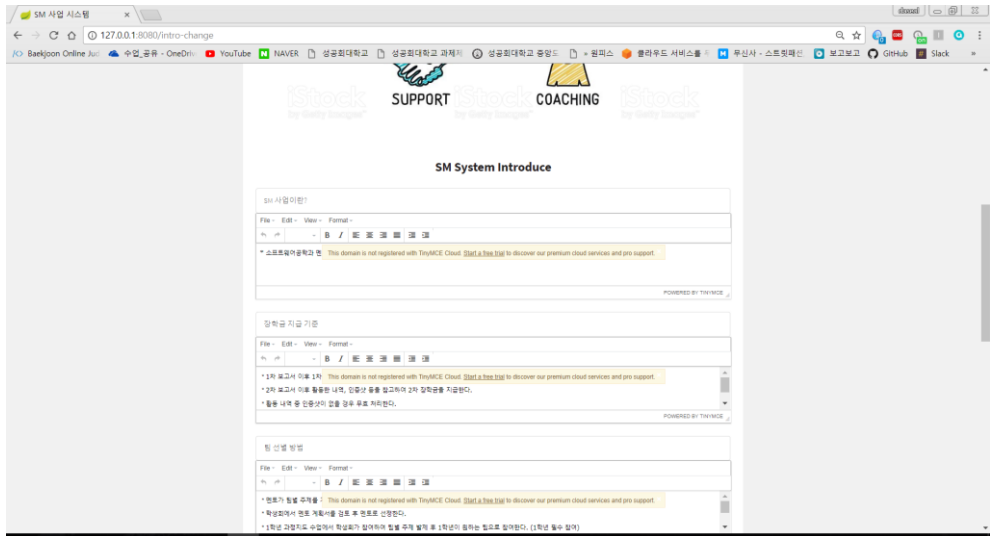
[그림 III-30] 회원 검색

시스템에 등록된 모든 회원을 검색하고 회원을 관리 할 수 있다.



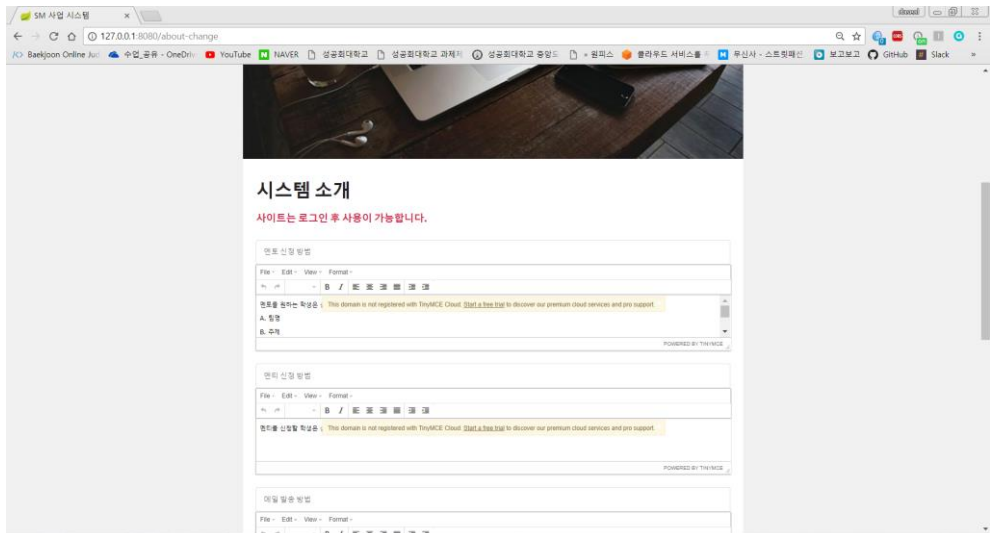
[그림 III-31] 회원 명단 업로드

엑셀 파일을 통해 한번에 회원 명단을 업로드한다. 서버는 엑셀 파일을 분석해 자동으로 회원가입을 진행한다. 이미 등록된 회원이 있으면 자동으로 등록하지 않는다.



[그림 III-32] Intro 수정

관리자는 SM 사업 소개 페이지 항목들을 소개할 수 있다.

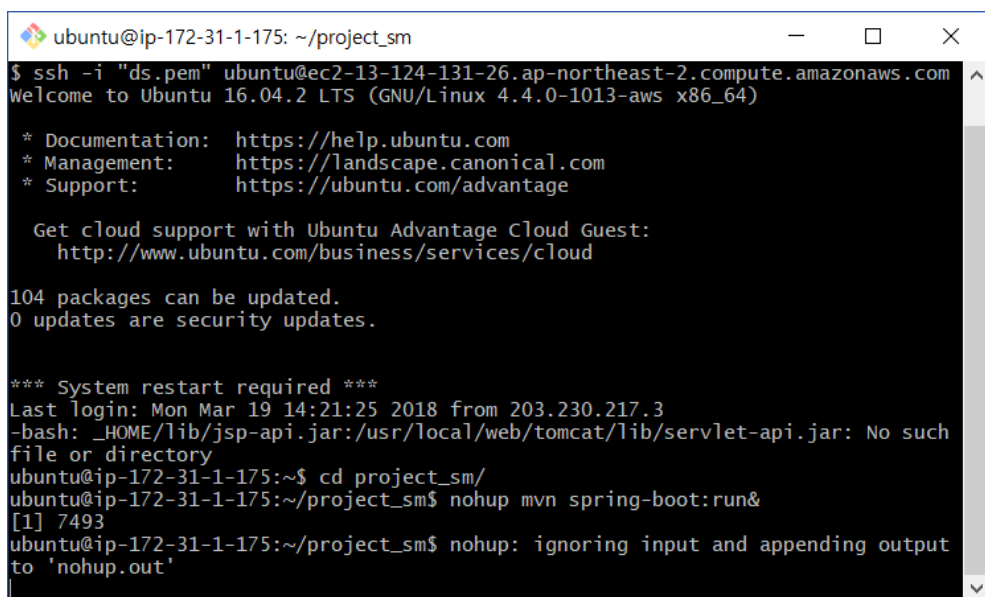


[그림 III-33] about 수정

관리자는 시스템 소개 페이지 항목을 수정할 수 있다.

3.3 서버

서버는 웹 서버 하나가 존재한다. Spring-boot를 사용해 개발했고, ORM 프레임워크로 Mybatis를 사용했다. Redis를 사용하기 위해 Spring Data Jpa를 추가했다.



```
ubuntu@ip-172-31-1-175: ~/project_sm
$ ssh -i "ds.pem" ubuntu@ec2-13-124-131-26.ap-northeast-2.compute.amazonaws.com
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-1013-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

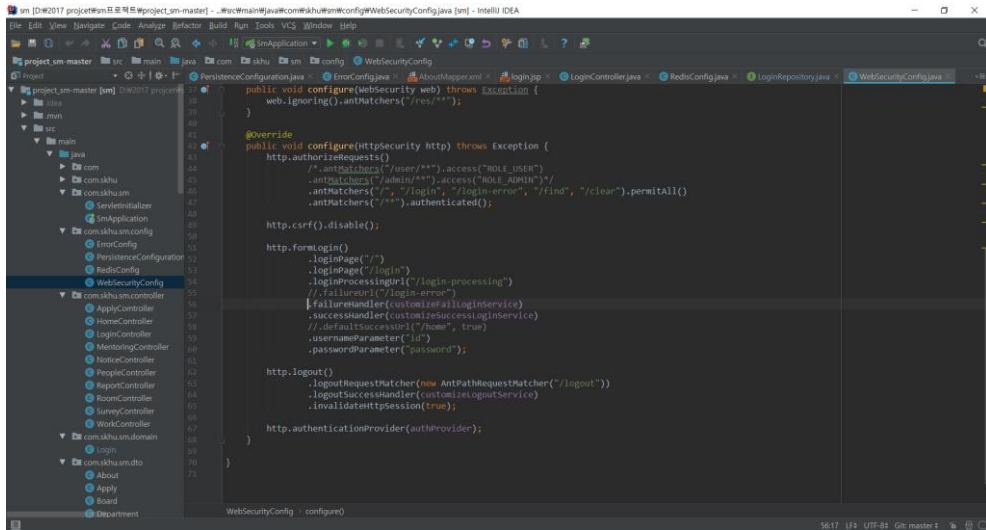
104 packages can be updated.
0 updates are security updates.

*** System restart required ***
Last login: Mon Mar 19 14:21:25 2018 from 203.230.217.3
-bash: _HOME/lib/jsp-api.jar:/usr/local/web/tomcat/lib/servlet-api.jar: No such
file or directory
ubuntu@ip-172-31-1-175:~$ cd project_sm/
ubuntu@ip-172-31-1-175:~/project_sm$ nohup mvn spring-boot:run&
[1] 7493
ubuntu@ip-172-31-1-175:~/project_sm$ nohup: ignoring input and appending output
to 'nohup.out'
```

[그림 III-34] 웹 서버 구동 화면

서버는 AWSEC2에서 구동이 되며 MVN으로 빌드하고 실행 한다. 백 그라운드 실행을 위해 nohup 명령어를 사용해 실행시킨다. 기본 포트인 8080을 사용하지만 포트 포워드를 통해 80으로 접속할 수 있다.

도메인 설정을 통해 bghgu.tk로 접속할 수 있다.



[그림 III-35] Spring Security

보안 설정을 위해 Spring Security 를 사용했고, 커스텀 핸들러를 사용했다.



[그림 III-36] 커스텀 핸들러

로그인이 실패 했을 경우 원인 표시를 위한 커스텀 핸들러이다.

```

# AWS key
cloud.aws.credentials.accessKey=
cloud.aws.credentials.secretKey=
cloud.aws.stack.auto=false
# AWS S3 bucket
cloud.aws.s3.bucket=project-sm
cloud.aws.region.static=ap-northeast-2

```

[그림 III-37] AWS 설정

```

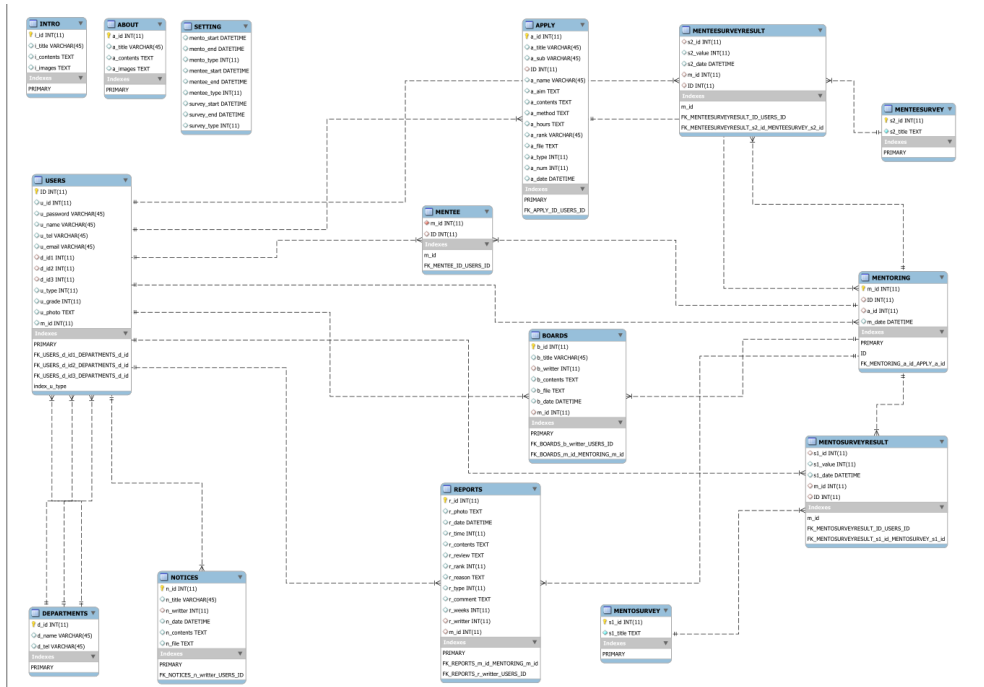
<!-- AWS -->
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-aws</artifactId>
</dependency>
</dependencies>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-aws</artifactId>
      <version>1.2.1.RELEASE</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>

```

[그림 III-38] AWS Dependency

AWS 제품 중 S3을 사용하기 위한 설정이다.



[그림 III-39]ERD 다이어그램

어떤 기능을 추가해도 감당이 될 수 있을 정도로 정규화 작업을 진행했다.
DB는 Mysql을 사용 했고,AWS RDS를 통해 배포되고 있다.

4. 참고문헌

Spring boot

<https://docs.spring.io/spring-boot/docs/2.0.0.RELEASE/reference/htmlsingle/>

Spring Security

<https://projects.spring.io/spring-security/>

Spring Data Redis

<https://projects.spring.io/spring-data-redis/>

Redis

<https://redis.io/documentation>

AWS RDS

<https://aws.amazon.com/ko/rds/getting-started/>

mysql

<https://dev.mysql.com/doc/>

Mybatis

<http://www.mybatis.org/mybatis-3/ko/index.html>

AWS S3

<https://aws.amazon.com/ko/s3/details/>

AWS EC2

<https://aws.amazon.com/ko/ec2/details/>

AWS ElastiCache

<https://aws.amazon.com/ko/elasticache/details/>