

# Mentor & Mentee

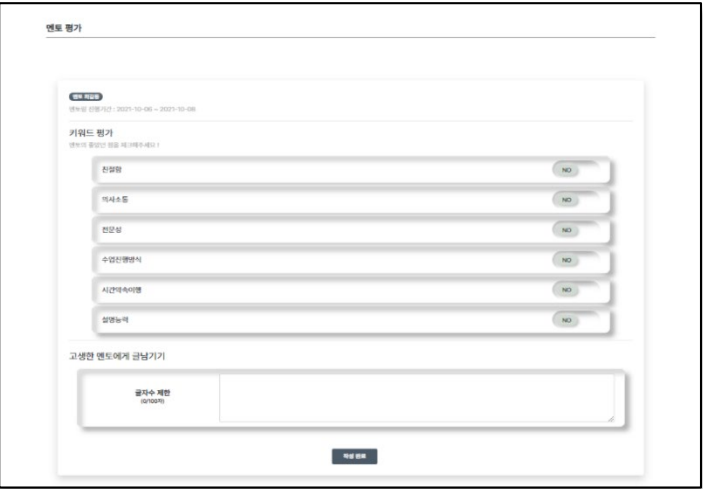
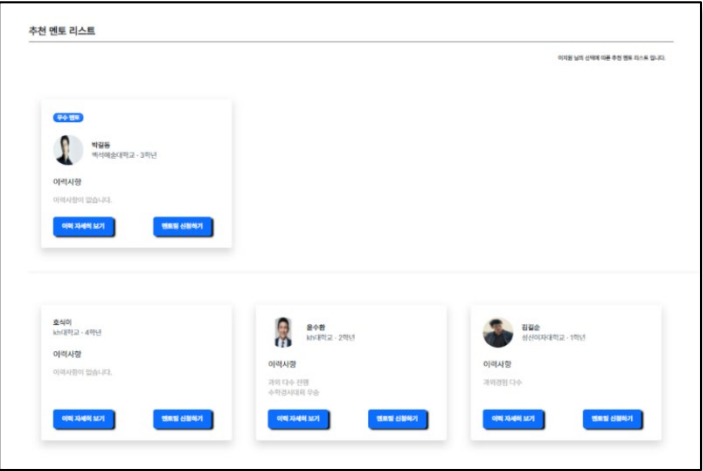
---

더 빠르게 높이 가다.

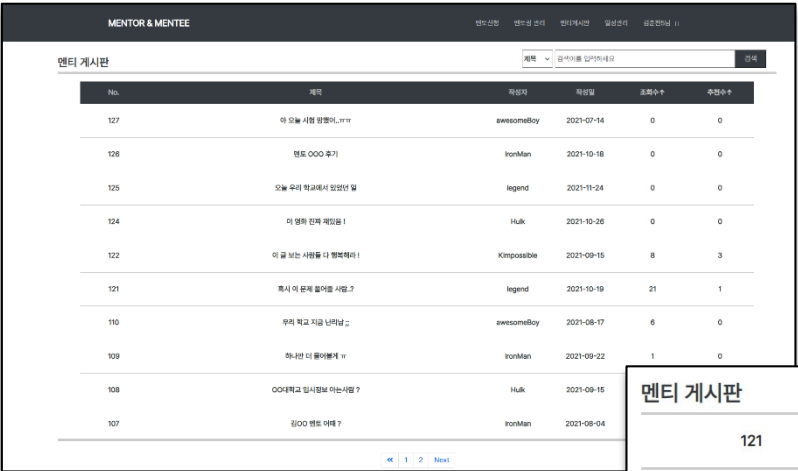
## About this Project

Mentor & Mentee Project는 중/고등학생과 대학생사이의 멘토링  
연결 공간을 제공 및 커뮤니티 사이트 입니다

멘토링연결공간제공



학생들간의정보공유커뮤니티



담당부분

저는 팀장을 맡아서 프로젝트를 진행했으며, 멘토링 시스템 구현을 맡았습니다.

M&M 프로젝트

코로나로 인해 대면으로 얻을 수 있는 정보(입시정보 등)를 얻지 못하는 학생들을 위해 대학생과의 멘토링 연결공간을 제공하고 학생들간의 커뮤니티 공간을 제공해주어 학생들의 학업에 도움을 주고자하는 사이트 입니다

멘티 게시판

121

혹시 이 문제 풀어줄 사람..?

21. 실수  $t$  에 대하여 곡선  $y=e^x$  위의 점  $(t, e^t)$ 에서의 접선의 방정식을  $y=f(x)$ 라 할 때, 함수  $y=|f(x)+k-\ln x|$ 가 양의 실수 전체의 집합에서 미분가능하도록 하는 실수  $k$ 의 최솟값을  $g(t)$ 라 하자. 두 실수  $a, b(a < b)$ 에 대하여  $\int_a^b g(t)dt = m$ 이라 할 때, <보기>에서 옳은 것만을 있는 대로 고른 것은? [4점]

<보 기>

ㄱ.  $m < 0$ 이 되도록 하는 두 실수  $a, b(a < b)$ 가 존재한다.  
ㄴ. 실수  $c$ 에 대하여  $g(c) = 0$ 이면  $g(-c) = 0$ 이다.  
ㄷ.  $a = \alpha, b = \beta(\alpha < \beta)$ 일 때  $m$ 의 값이 최소이면  $\frac{1+g'(\beta)}{1+g'(\alpha)} < -e^2$ 이다.

① ㄱ

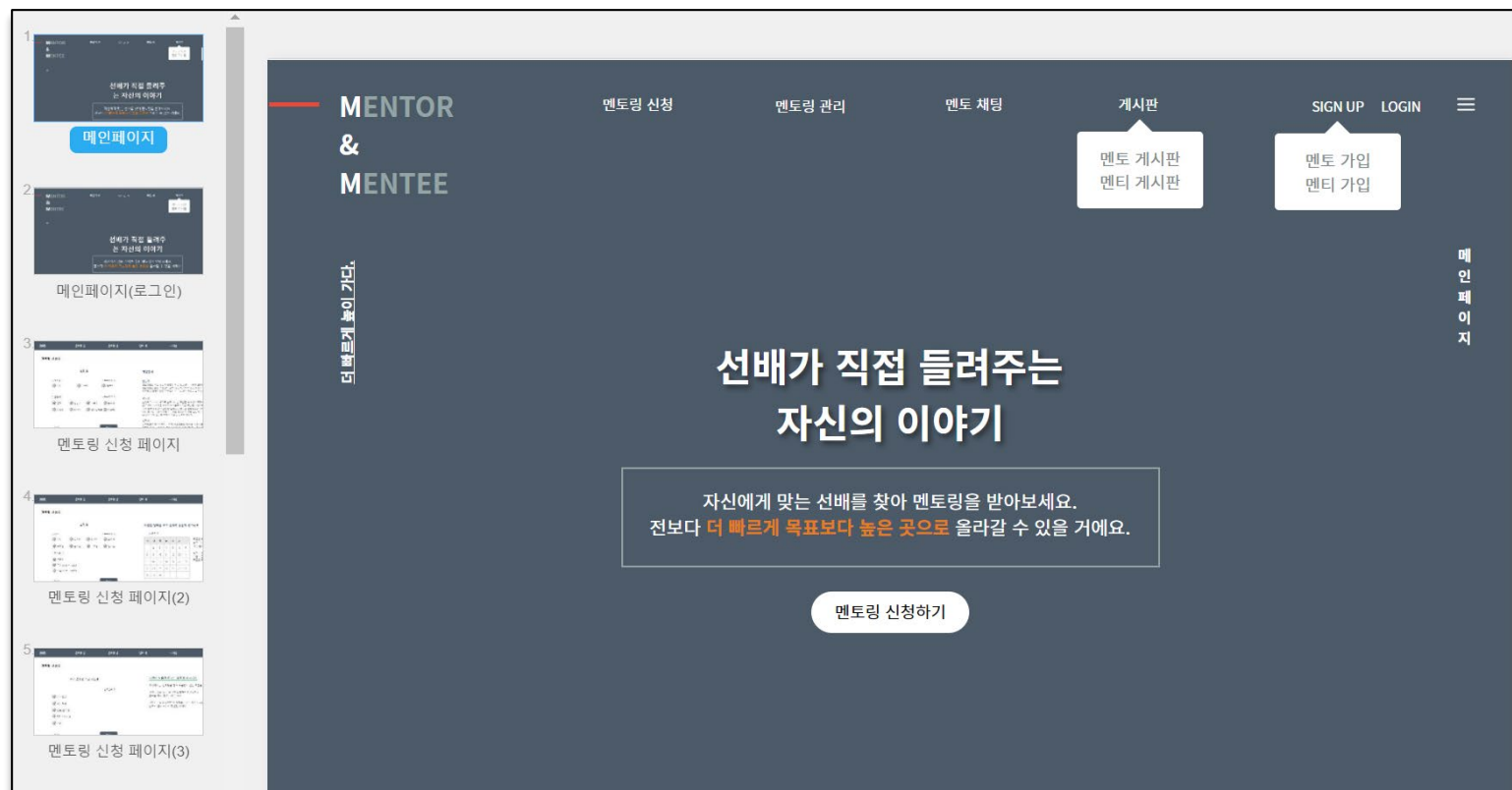
② ㄴ

③ ㄱ, ㄴ

④ ㄱ, ㄷ

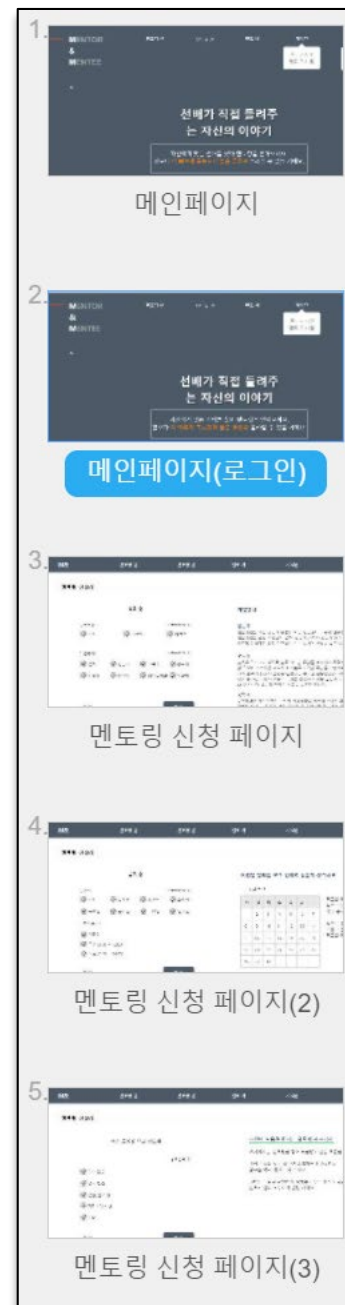
⑤ ㄱ, ㄴ, ㄷ

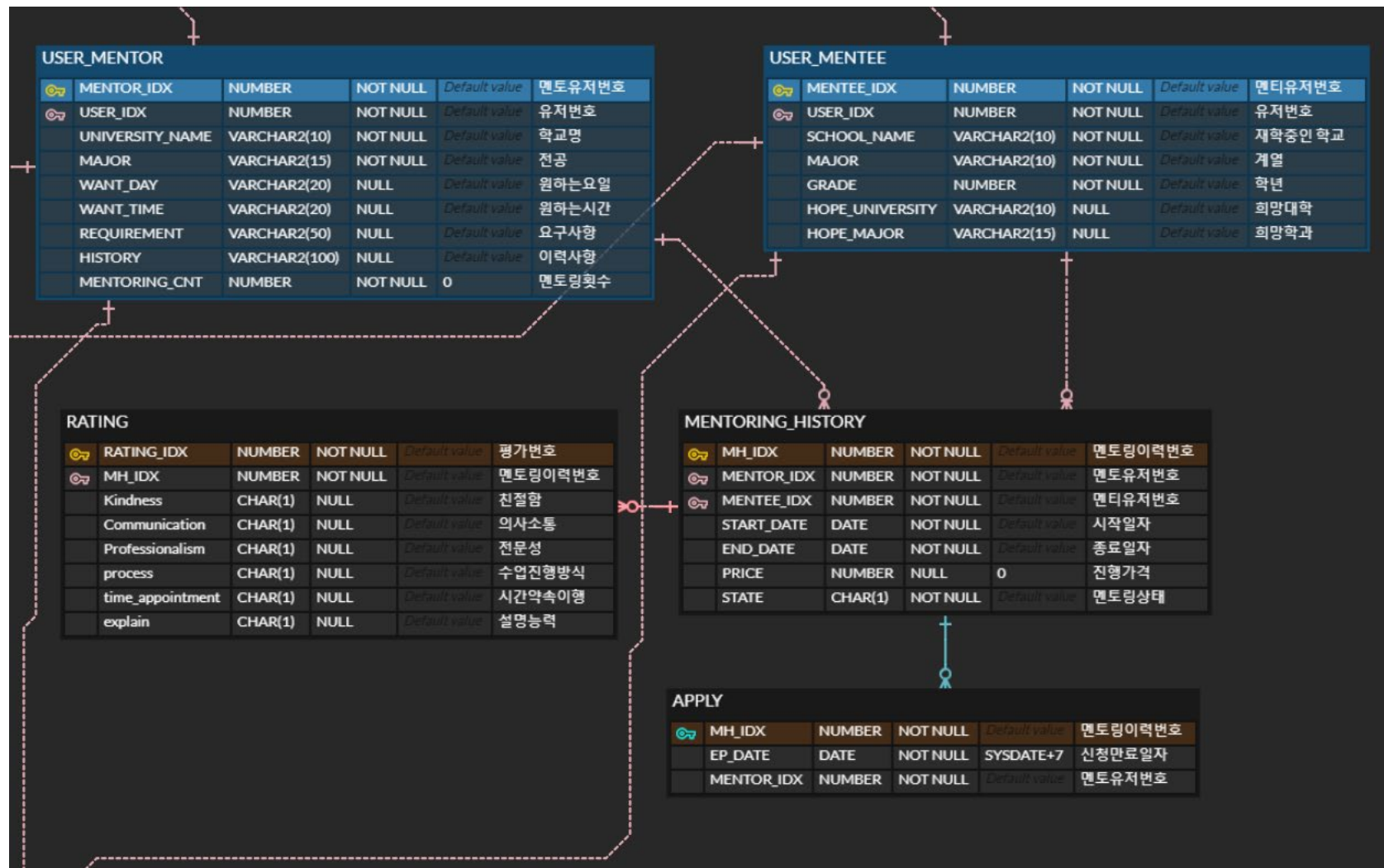
모의고사에서 나온 문제인데 혹시 풀이 알려줄 사람 있어?



## 카카오 오븐

카카오 오븐을 사용하여 각자 맡은 부분의 프로토타입을 만든 후 회의를 통해 수정하였습니다.





## ERD CLOUD

DB 모델링은 담당 파트만 캡처한 부분입니다.  
DB 설계에는 팀원들과 동시에 작업이 가능하고  
공유도 편하기 때문에 ERD CLOUD를 사용하였습니다.

신청 중인 멘토링과 진행 중인 멘토링을 따로 구분하기  
위해 멘토링 상태가 진행 중일 때 사용할 APPLY 테이블을  
따로 만들었습니다.

Mentoring\_History 테이블은 User\_Mentor와 User\_Mentee  
테이블을 기본키를 외래키로 받아  
각 회원의 정보를 조인하여 사용하였습니다.

Rating 테이블은 이미 등록된 멘토링 내역인지 확인하고  
Mentor와 Mentee의 정보를 가져오기 위해  
Mentoring\_History 테이블의 기본키를  
외래키로 받아 사용하였습니다.

Front

JSP, CSS, JavaScript

Backend

Java

DataBase

Oracle

Server

Apache Tomcat v8.5

Tool

Github, Eclipse, git

Library & API

Kakao API (Kakao Login)

Jersey (Toss API)

Zxing (QR코드 생성)

Quartz (스케줄러)

FullCalender (일정관리)

Gson

Cos

### 희망 학과 선택

---

1. 학교 유형

☐ 상관없음      ☐ 대학교      ☐ 전문대

---

2. 전공 계열 \*중복선택가능

☐ 상관없음    ☐ 인문계열    ☐ 교육계열    ☐ 공학계열

☐ 사회계열    ☐ 자연계열    ☐ 예체능계열    ☐ 의약계열

1/3 다음

### 희망 요일 및 시간 선택

---

1. 원하는 요일 \*중복선택가능

☐ 상관없음    ☐ 월요일    ☐ 화요일    ☐ 수요일

☐ 목요일    ☐ 금요일    ☐ 토요일    ☐ 일요일

---

2. 원하는 시간

☐ 상관없음

☐ 오전(09:00 ~ 12:00)

☐ 오후(12:00 ~ 17:00)

☐ 저녁(17:00 ~ 21:00)

2/3 이전    다음

### 원하는 진행 방식 선택

---

☐ 상관없음

☐ 비대면(화상채팅)을 원해요

☐ 저희 동네로 와주세요

☐ 멘토의 동네로 갈게요

☐ 카페/스터디룸 등 대여 공간을 원해요.

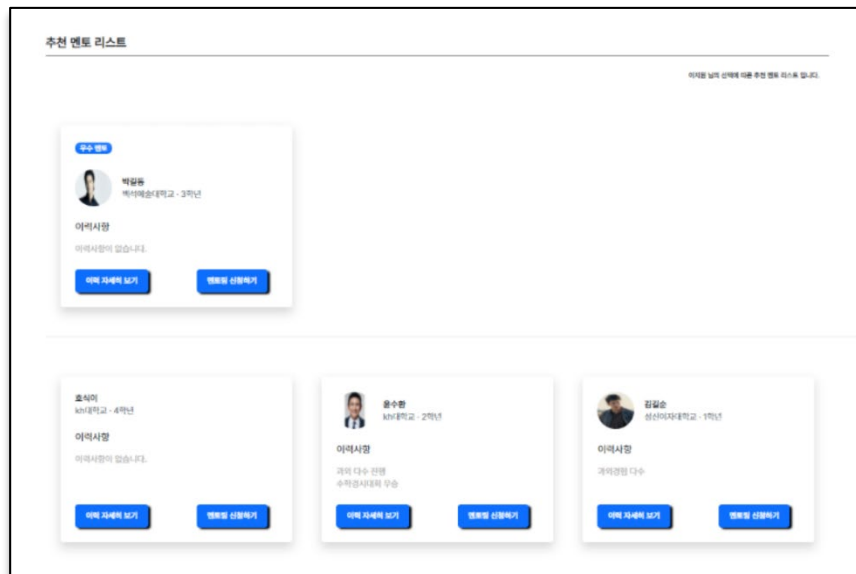
3/3 이전    작성완료

## Choose Condition

원하는 조건에서 멘토링을 할 수 있도록 사용자에게 선택지를 주고 그에 부합하는 멘토 리스트를 사용자에게 제공하도록 구현하였습니다

사용자에게 원하는 멘토의 조건을 form으로 받고 선택에 따라서 동적 query문을 만들어 DB에 요청합니다

## 멘토리스트페이지



## Mentor List

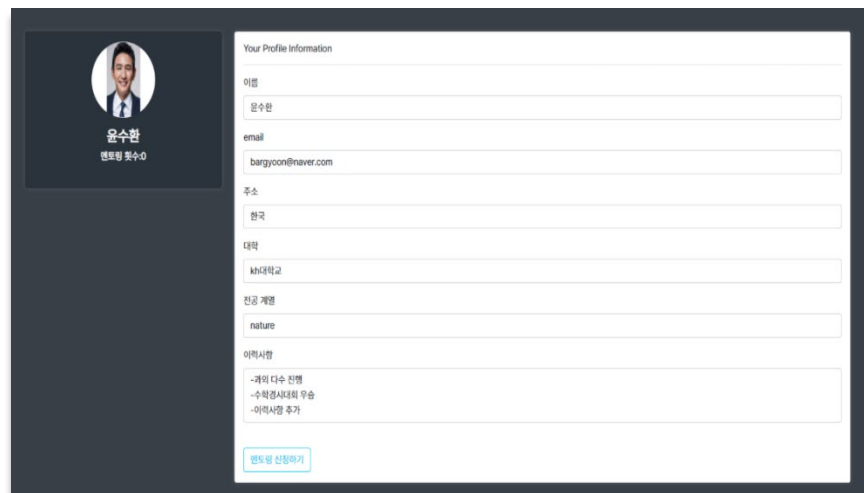
사용자가 선택 조건에 부합하는 멘토 리스트를 DB에서 가져온 후 화면에 출력하였습니다.

Mentor의 Role을 MO00(일반 멘토) MO01(우수 멘토)로 나누어서 DB에서 가져온 목록을 2개의 List로 나누어서 View단으로 전달하고 우수 멘토는 위쪽에 일반 멘토는 아래쪽에 구분되어서 표시되도록 하였습니다.

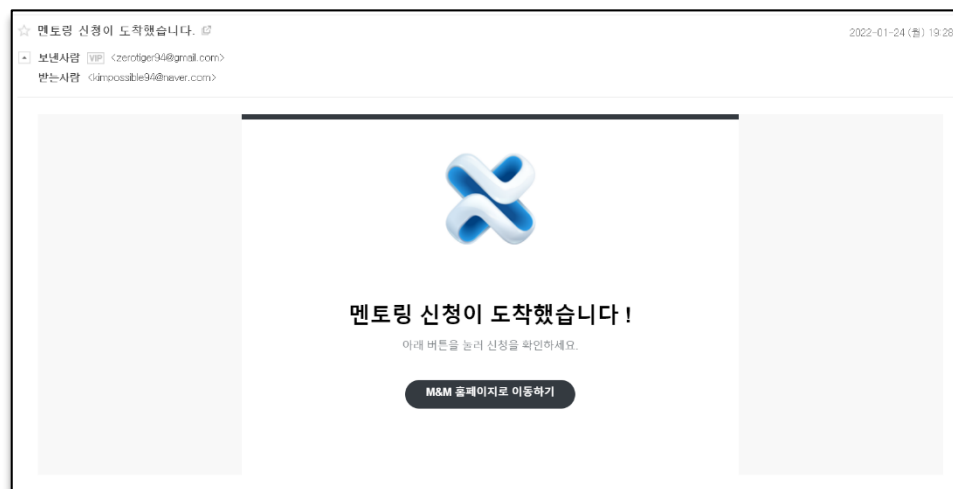
멘토 리스트에서 '이력서 자세히 보기' 버튼으로 멘토에 대한 정보를 좀 더 자세히 볼 수 있는 페이지로 이동할 수 있습니다.

'멘토링 신청하기' 버튼을 누르면 해당 멘토에게 Gmail SMTP 프로토콜을 사용해 알림 메일을 보내고 멘토링 신청 내역에 등록됩니다.

## 멘토정보상세페이지



## 멘토링신청알림메일



## 멘토링관리(멘토)

멘토 최길동님

환영합니다

신청 받은 멘토링

2건

진행 중인 멘토링

1건

완료한 멘토링

2건

신청중인 멘토링

이지원

멘토링 신청 날짜 : 2021-10-05

재신청 횟수 : 0/2

멘토 수락

박길자

멘토링 신청 날짜 : 2021-10-05

재신청 횟수 : 0/2

멘토 수락

\* 신청일 기준 3일 내로 멘토의 수락이 없을 시 삭제됩니다.

## 멘토링관리(멘티)

멘티 이지원님

환영합니다

신청 중인 멘토링

1건

진행 중인 멘토링

1건

완료한 멘토링

2건

완료한 멘토링

박길동

멘토 진행 기간 : 2021-10-06 ~ 2021-10-14

재신청

평가하기

박길동

멘토 진행 기간 : 2021-10-04 ~ 2021-10-21

재신청

평가하기

\* 최대 1년 내의 멘토링 기록만 보여줍니다.

## Manage Mentoring

멘토링 관리 페이지 입니다.

멘토링 관리페이지는 JSP를 사용해 같은 페이지 안에서 session에 저장되어 있는 회원의 정보에 따라 다르게 보이도록 구현하였습니다.

신청받은 멘토링에서 멘토(위쪽 사진)는 멘토링 수락 버튼을 멘티(아래쪽 사진)은 재신청과 결제하기 버튼이 보여지도록 하였습니다.



## 멘토링수락페이지

추가 입력 사항

멘토링 진행을 위해 입력해야하는 항목입니다. (\* 가격 제외)

추가 입력 항목

1. 희망 시작 일자

연도-월-일

2. 희망 종료 일자

연도-월-일

3. 진행 가격

0

작성 완료

## Mentoring Accept

멘토링 관리 페이지에서 이동하는  
멘토링 수락 페이지 입니다

멘토링 수락 페이지에서는 멘토가 멘티와 협의한 가격,  
멘토링 기간을 직접 입력하도록 하였습니다  
Servlet Filter를 활용해 멘토만 접근이 가능하도록  
구현하였습니다

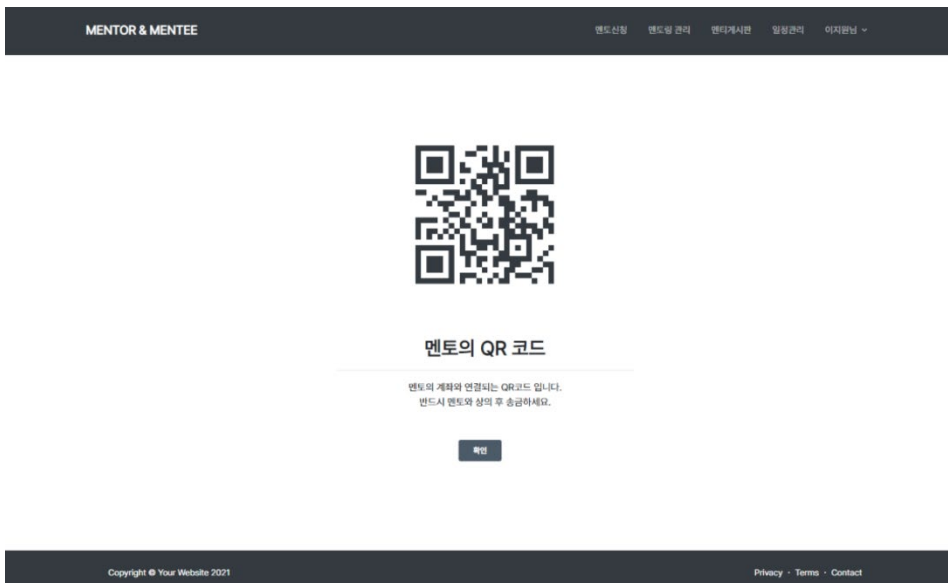
추가 입력 사항을 전부 다 작성하면 Apply 테이블에서  
삭제되고 상태가 진행 중으로 바뀌어 DB에 등록됩니다.

## 멘토링결제페이지

## Payment

멘토링 결제 페이지 입니다.

유저 간의 거래를 구현하고 싶었으나 구현에 한계가 있어  
Toss api를 활용하여 멘토의 계정에 입력되어 있는  
계좌 정보를 이용해 toss app과 연결되는 link를 생성하고  
이 링크를 zxing 라이브러리를 사용해 QR코드로 만들어  
페이지에 보여줘 멘티 유저가 스캔을 통해 결제하도록  
하였습니다



```

public void contextInitialized(ServletContextEvent sce) {
    try {
        // Grab the Scheduler instance from the Factory
        Scheduler scheduler = StdSchedulerFactory.getDefaultScheduler();

        Calendar calendar = Calendar.getInstance();

        calendar.set(Calendar.HOUR, 0);
        calendar.set(Calendar.MINUTE, 0);
        calendar.set(Calendar.SECOND, 0);

        Date date = calendar.getTime();
        JobDetail job = newJob(JobDelete.class).withIdentity("deleteFromDB", "group1").build();

        Trigger trigger = newTrigger()
            .withIdentity("deleteTrigger", "group1").startNow()
            .withSchedule(simpleSchedule()
                .withIntervalInHours(24)
                .repeatForever()
                .startAt(date)
                .build());

        scheduler.scheduleJob(job, trigger);
        scheduler.start();

    } catch (SchedulerException se) {
        se.printStackTrace();
    }
}

```

## Quartz Scheduler

멘토링 신청 만료일자와 멘토링 진행 기간이 지난 항목들에 대해서 상태를 매일 일괄적으로 바꿔주는 기능을 구현하고 싶어서 찾아보다 Scheduler를 알게되어 적용하였습니다.

ServletContextListener에 Scheduler를 등록하여 어플리케이션이 실행될 때 같이 시작되어 매일 0시 정각에 기간이 지난 데이터에 대해서 일괄적으로 상태를 바꿔주도록 구현하였습니다.

```

public class JobDelete implements Job {

    @Override
    public void execute(JobExecutionContext context) throws JobExecutionException {
        MentoringDao mDao = new MentoringDao();
        //apply_history에서 삭제
        mDao.deleteAh();
        //mentoring_history에서 삭제
        mDao.deleteMh();
        //mentoring_history에서 기한이 지난 멘토링 상태 변경
        mDao.updateMh();
    }
}

```

## 멘토링평가페이지

멘토 평가

멘토링 기간

멘토링 진행기간 : 2021-10-06 ~ 2021-10-08

키워드 평가

멘토의 좋았던 점을 체크해주세요 !

친절함	NO
의사소통	NO
전문성	NO
수업진행방식	NO
시간역속이행	NO
설명능력	NO

고생한 멘토에게 글남기기

글자수 제한  
(0/100자)

작성 완료

## Mentor Review

멘토 평가 페이지입니다.

멘토링 기간이 끝난 멘토링에 대해서 멘토링 관리 페이지에서 멘토링 평가를 남길 수 있습니다.

6가지의 키워드로 평가하고 멘토에게 글을 남길 수 있습니다.  
여기에서 멘티가 남긴 평가들을 종합하여 우수 멘토와 일반 멘토로 나눕니다.

Review 테이블에 진행한 멘토링 번호를 남겨두어 하나의 멘토링에 하나의 평가만 등록할 수 있도록 하였습니다.

# Mentor & Mentee

---

더 빠르게 높이 가다.

읽어주셔서 감사합니다  
잘 부탁드립니다