

DOUM

4->3조

20160982 이슬기

20160938 김예진

20160984 이예건

01

시스템 개요

01-1.시스템 개요

-검색

회원은 팀 목록을 검색할 수 있고 팀 정보를 통해 자신이 원하는 팀에 참여할 수 있다.

-등록

멤버로 가입한 회원은 자신이 종사하는 분야에 멘토로 등록할 수 있다.
멘토는 새로운 팀을 개설할 수 있다.

-추천

회원은 회원가입 시 체크한 선호 분야로 멘토를 추천 받을 수 있다.

01-2. 시스템 특징

- 청소년, 대학생을 대상으로 한 맞춤 사이트
- 멘토의 자기PR을 보고 멘티들이 선택적으로 신청할 수 있다.
- 멘토링 가입시 기입한 사항을 토대로 멘토-멘티 자동추천을 받을 수 있다.
- 구인 글을 조건 별 검색을 가능하게 구현하여 회원에게 필요한 멘토링을 쉽게 찾을 수 있다.
- 사이트를 통한 실제 멘토링 경험자의 후기를 통해 멘토링 구인에 있어 도움을 받는다.

02

구현 기능

02. 구현기능

핵심 기능

1. 팀 관련 기능

- 멘토링 팀(TeamSmall)
 - 팀 개설하기
 - 팀 목록보기
 - 팀 상세정보 확인
 - 팀 삭제하기
- 특강(TeamLarge)
 - 특강 개설하기
 - 특강 목록보기
 - 팀 상세정보 확인

2. 멘토 관련 기능

- 멘토 등록하기
- 멘토 목록보기
- 장소 대여신청
- 멘티 승인하기

3. 회원 관련 기능

- 로그인 및 회원가입
- 후기 게시글 작성
- 마이페이지 확인
- 팀 활동 기록 등록

02. 구현기능

세부 기능

1. 멘토링 팀 관련 기능

- 팀 개설하기
 - 멘토 등록을 해야 개설 메뉴 진입 기능
 - 활동 분야 선택
- 팀 목록보기
 - 멘티 4명이 모이면 모집 완료된 팀에서 확인 기능
 - 멘티 수가 4명보다 적으면 모집중인 팀에서 확인 기능
 - 전체 팀 목록에서 팀 이름 클릭하면 해당 팀 상세정보 확인 기능
- 팀 상세정보 확인
 - 팀의 멘티 이름, 활동분야 등 상세정보 확인 기능
 - 팀의 멘티 등록하기 기능
 - 멘티 승인을 위한 대기 목록 확인 기능
 - 해당 팀의 멘토일 때 멘티 신청하기 버튼 보이지 않는 기능
- 팀 삭제하기

02. 구현기능

세부 기능

2. 특강 관련 기능

- 특강 개설하기
 - 멘토 등록을 해야 개설 메뉴 진입 가능
 - 활동 분야 선택
 - 제한 인원 선택
- 특강 목록보기
 - 특강 제한 인원이 모이면 모집 완료된 팀에서 확인 가능
 - 특강 제한 인원보다 신청인원이 적으면 모집중인 팀에서 확인 가능
 - 전체 특강 목록에서 특강 이름 클릭하면 해당 특강 상세정보 확인 가능
- 특강 상세정보 확인
 - 특강의 멘토 이름, 활동분야 등 상세정보 확인 가능
 - 특강의 수강생 등록하기 기능
 - 수강생 등록하면 특강의 상세정보에서 신청인원 수 증가 기능

02. 구현기능

세부 기능

3. 멘토 관련 기능

- 멘토 등록하기
 - 자신의 활동분야 선택
- 멘토 목록보기
 - 활동분야에 따른 멘토 검색 기능
 - 자신의 활동분야 선택
 - 전체 멘토 목록에서 멘토 이름 클릭하면 멘토 상세정보 확인기능
- 장소 대여신청
 - 장소 목록 확인 기능
 - 멘토링 팀 활동 장소 목록 보기
 - 멘토링 팀 활동 장소 지점 클릭하면 장소 상세보기 기능
 - 멘토링 팀 활동 장소 대여 신청 기능
 - 멘토링 활동할 팀 고르고 장소 신청
- 멘티 승인하기
 - 멘토로 활동하는 멘토링 팀에 멘티 승인 대기자를 승인해주는 기능

02. 구현기능

세부 기능

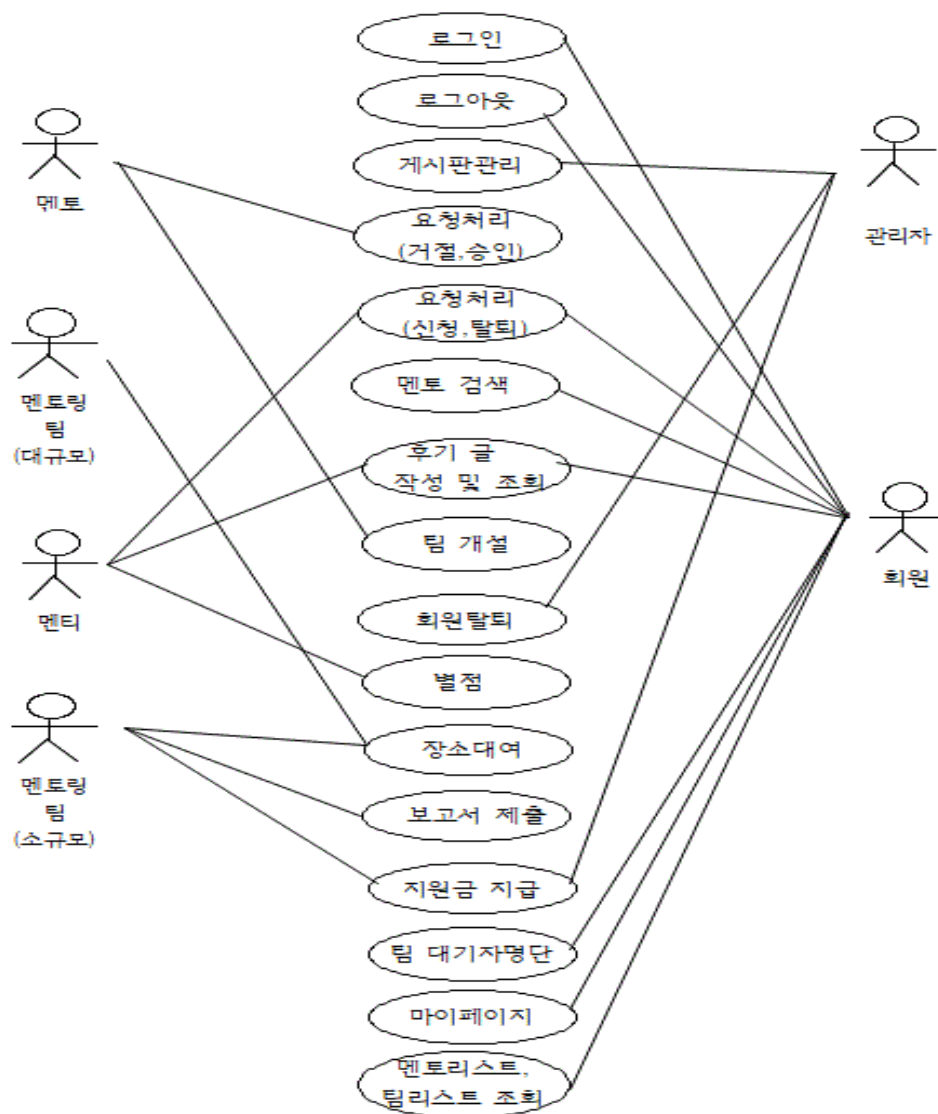
4. 회원 관련 기능

- 로그인 및 회원가입
 - 후기 게시글 작성
 - 자신이 참여했던 팀 및 특강의 멘토 후기 게시글 등록 기능
- 마이페이지 확인
 - 자신의 상세정보 확인 기능
 - 자신이 멘티로 참여했던 팀 및 특강 목록 확인 기능
 - 자신이 멘토로 참여했던 팀 및 특강 목록 확인 기능
 - 팀 및 특강의 이름을 누르면 상세정보 페이지로 이동 기능
- 멘토링 팀 활동 기록 등록
 - 활동 보고서 목록 확인 및 업로드 기능
 - 활동비 영수증 첨부 및 사용 금액 입력 기능
 - 처음 활동 보고서가 업로드 되면 지원금 지급 기능
 - 활동비 입력 값으로 지원금 차감 기능
- 멘토링 팀 및 특강 신청하기 기능
- 팀 탈퇴 기능
- 회원가입 때 선택한 관심분야 별 멘토 추천 기능
 - 후기 게시글들에 등록한 별점을 토대로 획득 별점이 높은 상위 5명 추천 기능

03

요구사항 분석 결과

03. 요구사항 분석 결과



actor	semantics
멘토	멘토로 등록한 회원이다.
관리자	후기게시판관리,장소관리,지원금관리 등 모든 분야의 관리자이다.
멘토링팀(소규모)	멘티들과 멘토로 구성된 모임이다. 최대 멘티 수는 4명이다.
멘토링팀(대규모)	세미나형식의 대형 강연을 들을 수 있는 팀이다. 최대 인원 수는 50명이다.
회원	사이트에 가입한 사람들이다.
멘티	소규모팀에 등록된 사람은 멘티가 된다.

use case	semantics
로그인	로그인하지 않으면 글을 볼 수 없다.
로그아웃	회원은 로그아웃할 수 있다.
게시판관리	관리자는 후기게시판을 관리한다.
요청처리(거절, 승인)	멘토는 회원의 요청에 답한다.
요청처리(신청, 탈퇴)	회원은 멘토링 팀에 신청, 멘티는 탈퇴 할 수 있다.
필드별 멘토 검색	회원들은 활동분야별로 멘토들을 검색할 수 있다.
후기 글 추가,수정,삭제 및 조회	회원은 멘토링을 함께한 타회원에 대한 후기를 추가,수정,삭제할 수 있다. 또한 회원은 후기 게시판의 글을 조회할 수 있다.
관리자 후기글삭제	관리자는 후기 게시판의 글과 댓글을 삭제할 수 있다.
후기 글 검색	회원은 검색 기준에 따른 글을 찾을 수 있다.
회원탈퇴	회원은 서비스를 이용하지 않을 시 회원탈퇴를 할 수 있다.
장소대여	장소를 관리하는 관리자는 스터디룸을 제공한다. 소규모/대규모팀은 스터디룸을 대여받을 수 있다.
별점	멘토링이 끝난 후 멘토에 대해 평가할 수 있다.
보고서 제출	멘토링팀(소규모)은 수행일지(보고서)를 작성한다.
지원금 지급	멘토링팀(소규모)은 지원금을 지급받는다. 관리자는 지원금을 관리한다.
팀 개설	멘토의 권한으로 소규모/대규모 멘토링 팀을 등록할 수 있다.
팀 대기자명단	회원은 가입을 희망하는 팀에 승인신청을 할 시 팀의 대기자명단에 등록되어 멘토의 요청을 기다린다.
마이페이지	회원은 자신의 정보를 조회할 수 있다. 또한 자신이 멘티로 활동하는 팀과 멘토로 활동하는 팀을 조회할 수 있다.
멘토리스트, 팀리스트 조회	회원은 현재 활동중인 멘토들의 리스트와 팀리스트를 조회할 수 있다.

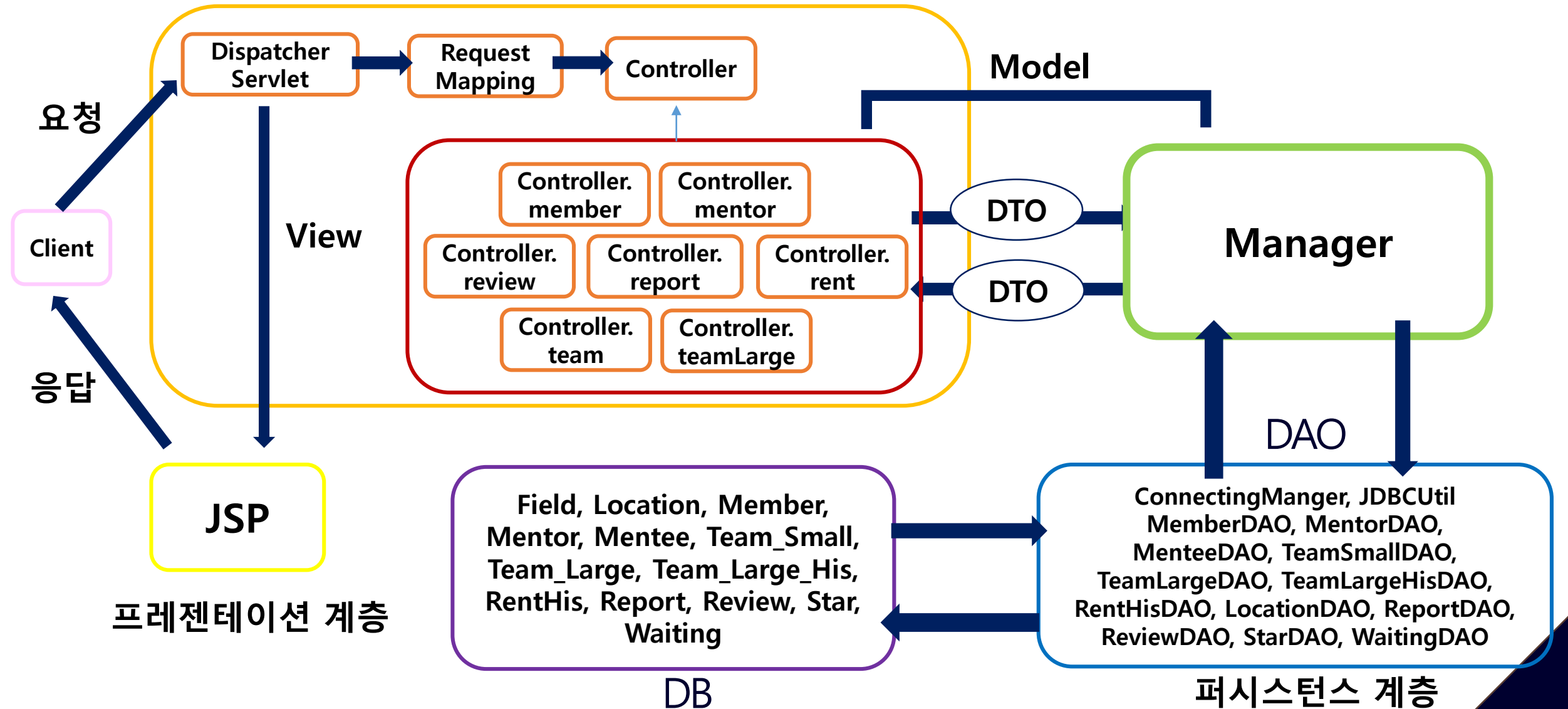
04

시스템 구성도

04. 시스템 구성도

MVC구조

Controller



05

UI 설계

05. UI설계

UI 구성

- WebContent
 - css
 - menu.css
 - user.css
 - images
 - logo.png
 - mypage_logo.png
 - js
 - jquery-1.12.4.min.js
 - main
 - mainPage.jsp
 - member
 - loginForm.jsp → 로그인 폼
 - memberJoin.jsp → 회원가입 폼
 - mentor
 - mentorDetail.jsp → 멘토의 상세정보 출력(이름, 활동분야, 자기소개) -> 멘토리스트에서 이름 클릭시
 - mentorJoin.jsp → 멘토로 등록하는 폼
 - mentorList.jsp → 멘토들의 리스트 출력 -> Ajax로 활동분야별 멘토검색 가능
 - META-INF
 - myPage
 - myPage.jsp → 회원 상세정보 페이지
 - rent
 - RentPlaceDetail.jsp → 장소대여내역 상세정보 페이지
 - rentPlacePage.jsp → 장소들의 목록 보는 페이지
 - RentPlaceRequest.jsp → 장소대여신청 하는 페이지



▲logo



▲mypage_logo

05. UI설계

UI 구성

- ▼ report
 - reportJoin.jsp → 보고서 등록 페이지
 - reportList.jsp → 올린 보고서들의 리스트
- ▼ review
 - reviewDetail.jsp → 후기게시판 게시물 상세보기
 - reviewJoin.jsp → 후기게시판에 게시물 작성
 - reviewList.jsp → 후기게시판 게시물 목록
- ▼ team
 - myTeamDetail.jsp → 내가 참여한 팀의 상세정보 보는 페이지
 - teamSmallDetail.jsp → 팀 상세정보 보는 페이지
 - teamSmallJoin.jsp → 팀 개설 폼 (멘토 권한)
 - teamSmallList.jsp → 팀 목록 -> Ajax로 모집중/모집완료된 팀 확인가능
 - teamSmallRequest.jsp → 팀 가입 요청하는 페이지
- ▼ teamLarge
 - teamLargeDetail.jsp → 특강 상세정보 보는 페이지
 - teamLargeJoin.jsp → 특강 개설 폼
 - teamLargeList.jsp → 특강 목록 -> Ajax로 모집중/모집완료된 팀 확인가능
- ▼ WEB-INF
 - > lib
 - web.xml
 - index.jsp
 - pom.xml

05. UI설계

aa (로그아웃)

<로그인>

사용자ID	<input type="text"/>
비밀번호	<input type="password"/>

▲ loginForm.jsp

Team

Mentor

장소대여 신청

후기게시판

관심분야 멘토추천

필드이름:IT

멘토이름 (ID)	별점	멘토 활동경력 횟수
김동덕 (cc)	7.0	5
김미미 (aa)	0.0	0

필드이름:수학

멘토이름 (ID)	별점	멘토 활동경력 횟수
숨숨이 (somsom)	2.0	0
더블 (ww)	0.0	1
이디에 (edf)	0.0	3
비씨디 (bcd)	0.0	0

필드이름:기계

멘토이름 (ID)	별점	멘토 활동경력 횟수
김수수 (tt)	4.0	0

▲ mainPage.jsp

05. UI설계



◀ myPage.jsp

사용자 관리 - 상세정보 보기

사용자ID	cc
이름	김동덕
패스워드	11
성별	woman
닉네임	르르
전화번호	01012345789
생년월일	180101
이메일	cc@naver.com
관심분야	IT, 수학, 기계

멘토링 팀 멘티 활동 내역

팀이름

멘토링 팀 멘토 활동 내역

팀이름
teamcc08
teamcc3
teamcc15
teamcc50
TeamKim

멘티 승인 대기 팀 목록

팀이름

특강 참여 내역

팀이름
t1

특강 멘토 활동 내역

팀이름
teamLargecc
teamlargecc1
teamLarge01

돌아가기

teamcc3팀 상세정보

teamcc3팀의 남은 지원금은 500000원입니다.

팀이름	teamcc3
멘토이름	김동덕
활동분야	IT
만나는위치	강원
멘티	더블, 이디에

팀 삭제 멘티 팀 탈퇴 보고서 리스트 보고서 리스트등록

▲ myTeamDetail.jsp

보고서 리스트

순서	최종 수정일	보고서URL	영수증URL
1	2018/12/27	Gant Chart.xlsx	Lauv - Paris in the rain.txt

이전 페이지로 돌아가기

▲ reportList.jsp

보고서 등록

보고서URI	<input type="button" value="파일 선택"/> 선택된 파일 없음
영수증URI	<input type="button" value="파일 선택"/> 선택된 파일 없음
사용금액	<input type="text"/>

보고서 등록

목록

이전 페이지로 돌아가기

▲ reportJoin.jsp

팀 리스트

☐ 모집중인 팀
 ☒ 모집 완료된 팀

→ Ajax구현

모집 완료된 팀:

팀 이름	활동 분야	만나는 위치
------	-------	--------

전체 팀 리스트 :

팀 이름	활동분야	만나는위치
teamcc3	IT	강원
teamcc08	IT	서울
TeamKim	기계	부산
teamcc15	수학	서울
teamcc50	수학	서울
team99	수학	광주
teamm	디자인	월곡
mentoring	디자인	대전
teamDBP1	수학	광명

* TeamKim 팀 상세정보 *

팀이름	TeamKim
멘토이름	김동덕
활동분야	기계
만나는위치	부산
멘티	

- 팀 가입 승인 대기 -

대기명단
<input type="checkbox"/> 김르르
<input type="checkbox"/> 김미미

승인

팀 목록 보기

▲ teamSmallList.jsp

팀 개설

팀이름	<input type="text"/>
활동분야	<input checked="" type="radio"/> IT <input type="radio"/> 수학 <input type="radio"/> 기계 <input type="radio"/> 어문 <input type="radio"/> 마케팅 <input type="radio"/> 디자인 <input type="radio"/> 미디어
만나는 위치	<input type="text"/>
종료날짜	YYYY/MM/DD <input type="text"/>

팀 등록

팀 목록보기

▲ teamSmallDetail.jsp◀ teamSmallJoin.jsp

05. UI설계

팀 리스트

● 모집중인 팀 ● 모집 완료된 팀

→ Ajax구현

모집중인 팀:

팀 이름	활동 분야	만나는 위치
teamLarge01	IT	서울
t1	디자인	제주
teamLargecc	기계	서울
teamlargecc1	IT	서울

전체 팀 리스트 :

팀이름	활동분야	만나는위치
teamLarge01	IT	서울
t1	디자인	제주
teamLargecc	기계	서울
teamlargecc1	IT	서울

teamLarge01팀 상세정보

팀이름	teamLarge01
멘토이름	김동덕
활동분야	IT
만나는위치	서울
현재인원/ 제한인원	7/ 10

이전 페이지로 돌아가기

▲ [teamLargeDetail.jsp](#)

▲ [teamLargeList.jsp](#)

특강 개설

팀이름	<input type="text"/>
시작날짜	<input type="text"/>
활동분야	---선택--- ▼
만나는위치	<input type="text"/>
제한수	<input type="text"/>

특강 등록

특강 목록보기

◀ [teamLargeJoin.jsp](#)

05. UI설계

DOM

멘토 등록

활동분야	<input type="radio"/> IT <input type="radio"/> 수학 <input type="radio"/> 기계 <input type="radio"/> 어문 <input type="radio"/> 마케팅 <input type="radio"/> 디자인 <input type="radio"/> 미디어
자기소개	<div style="border: 1px solid #ccc; padding: 5px;"></div>

▲ mentorJoin.jsp

멘토 목록 검색

활동분야	<input checked="" type="radio"/> IT <input type="radio"/> 수학 <input type="radio"/> 기계 <input type="radio"/> 어문 <input type="radio"/> 마케팅 <input type="radio"/> 디자인 <input type="radio"/> 미디어
------	--

→ Ajax구현

멘토 이름	아이디	별점
김동덕	cc	7
김미미	aa	0

* 전체 멘토 리스트 *

이름(ID)	활동분야
김미미 (aa)	IT
바씨디 (bcd)	수학
김동덕 (cc)	IT
이디에 (edf)	수학
김규일 (q1)	미디어
김규비 (qtp)	어문
숨숨이 (somsom)	수학
김수수 (tt)	기계
더블 (ww)	수학

멘토 등록하기

▲ mentorList.jsp

멘토 상세정보 보기

이름	이디에
활동분야	수학
자기소개	비즈니스~

멘토 목록으로 돌아가기

▲ mentorDetail.jsp

05. UI설계

장소 리스트

위치이름
강남 1호점
강북 2호점
강서 3호점
성동 4호점
마포 5호점
종구 6호점
수원 7호점
구리 8호점
광명 9호점
고양 10호점
강릉 11호점
춘천 12호점
대전 13호점
인천 14호점
인천 14호점
광주 15호점
대구 16호점
부산 17호점
제주 18호점

장소 대여 내역 보기

지점 이름	강남 1호점
주소	서울특별시 강남구 삼성동 20-4

장소 대여 신청 지점 목록으로 돌아가기

1번방 대여내역 보기

팀이름	날짜	시간
teamcc08	2019/01/01	11:00

2번방 대여내역 보기

팀이름	날짜	시간
teamcc08	2019/01/13	10:00

3번방 대여내역 보기

팀이름	날짜	시간
-----	----	----

▲ RentPlaceDetail.jsp

▲ rentPlacePage.jsp

팀 개설

대여 장소 지정 이름	경기 수원 7호점
빌리려는 장소 번호	---선택---
장소 대여 등록할 팀	---선택---
날짜 입력	<input type="text"/>
시간 입력	---선택---

장소대여 신청 목록 취소

▲ RentPlaceRequest.jsp

05. UI설계

번호	제목	작성자	작성일	조회수
71	teamtt의review01	김동덕	2019/01/01	9
72	teamLarge01팀의review01	큐큐	2019/01/01	2
73	teamLarge01팀의review01	기요요	2019/01/10	1
74	teamtt의리뷰03	김동덕	2019/01/01	3
75	teamlarge01의리뷰11	큐큐	2019/01/01	2
76	hello	기요요	2018/11/12	2
77	cc	김수수	2019/01/01	2

리뷰 상세정보 보기

리뷰제목	hello
리뷰작성자	yy
리뷰에 대한 멘토	김동덕
리뷰내용	joah
리뷰날짜	2018/11/12
별점	만족성: 1 참여성: 1 준비성: 1
총 별점	1.0
조회수	3

[글쓰기](#)

▲ [reviewList.jsp](#)

후기글 작성

제목	<input type="text"/>
멘토id	<input type="text" value="---선택---"/>
후기 내용	<input type="text"/>
후기 날짜	<input type="text"/>
만족성	<input type="text" value="---선택---"/>
참여성	<input type="text" value="---선택---"/>
준비성	<input type="text" value="---선택---"/>

▲ [reviewDetail.jsp](#)

▲ [reviewJoin.jsp](#)

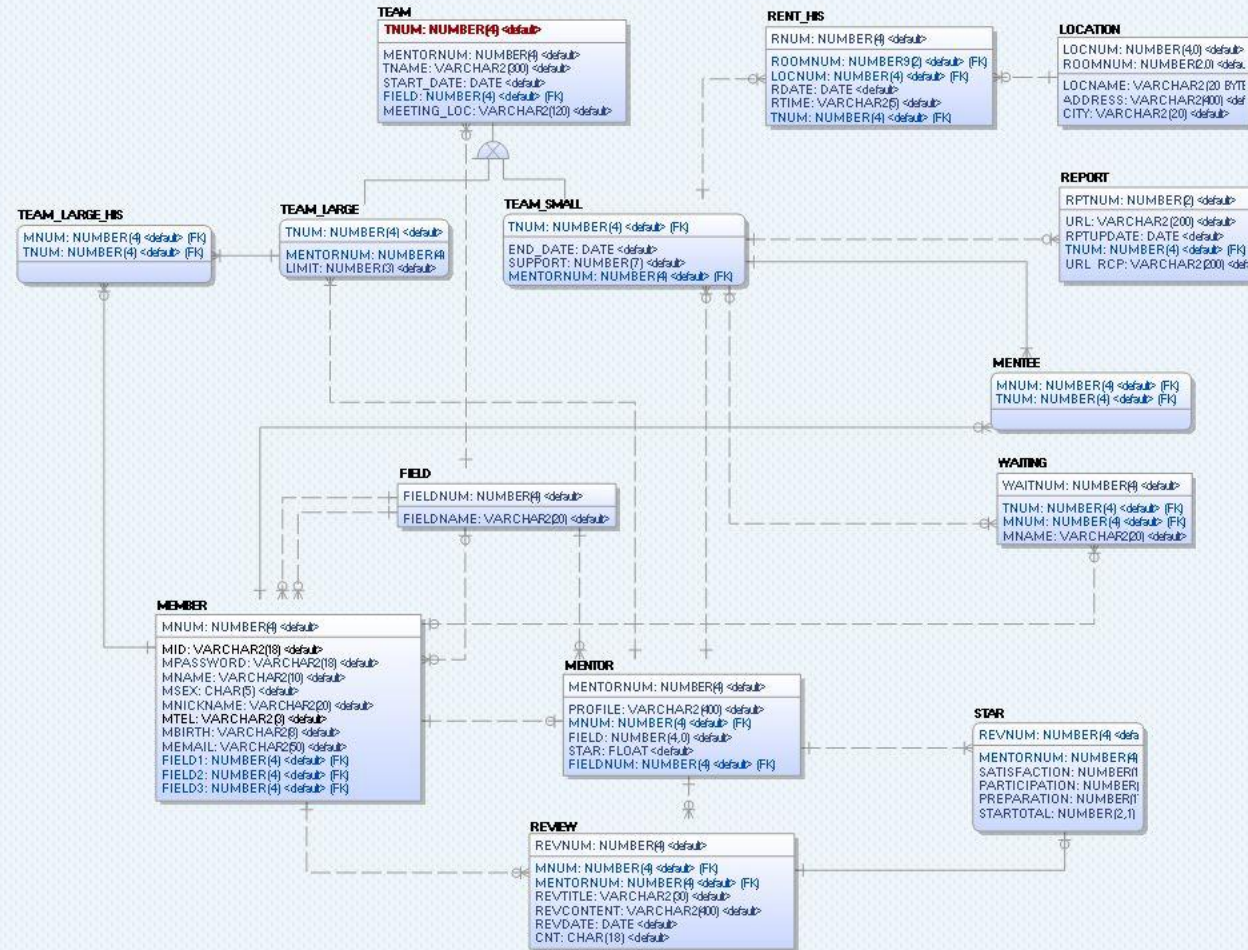
목록보기

06

데이터베이스 설계

06. 데이터베이스 설계

ER Diagram



06. 데이터베이스 설계

테이블 스키마

```
-- DDL for Table FIELD
-----
CREATE TABLE "DBP0103"."FIELD"
(
  "FIELDNUM" NUMBER(4,0),
  "FIELDNAME" VARCHAR2(20 BYTE)
)
-----
-- DDL for Index FIELD_PK
-----

CREATE UNIQUE INDEX "DBP0103"."FIELD_PK" ON "DBP0103"."FIELD" ("FIELDNUM")
-----
-- DDL for Index FIELD_UK1
-----

CREATE UNIQUE INDEX "DBP0103"."FIELD_UK1" ON "DBP0103"."FIELD" ("FIELDNAME")
-----
```

▲ FIELD TABLE

```
-- DDL for Table MEMBER
-----
CREATE TABLE "DBP0103"."MEMBER"
(
  "MNUM" NUMBER(4,0),
  "MID" VARCHAR2(18),
  "MPASSWORD" VARCHAR2(18),
  "MNAME" VARCHAR2(10),
  "MSEX" CHAR(5),
  "MNICKNAME" VARCHAR2(20),
  "MTEL" VARCHAR2(13),
  "MBIRTH" VARCHAR2(8),
  "MEMAIL" VARCHAR2(50),
  "FIELD1" NUMBER(4,0),
  "FIELD2" NUMBER(4,0),
  "FIELD3" NUMBER(4,0)
)
-----
-- DDL for Index XPK회원
-----

CREATE UNIQUE INDEX "DBP0103"."XPK회원" ON "DBP0103"."MEMBER" ("MNUM")
-----
-- DDL for Index MEMBER_UK1
-----

CREATE UNIQUE INDEX "DBP0103"."MEMBER_UK1" ON "DBP0103"."MEMBER" ("MID", "MNICKNAME")
-----
```

▲ MEMBER TABLE

```
-- DDL for Table LOCATION
-----
CREATE TABLE "DBP0103"."LOCATION"
(
  "ROOMNUM" NUMBER(2,0),
  "LOCNUM" NUMBER(4,0),
  "LOCNAME" VARCHAR2(20),
  "ADDRESS" VARCHAR2(400),
  "CITY" VARCHAR2(20)
)
-----
-- DDL for Index XPK장소
-----

CREATE UNIQUE INDEX "DBP0103"."XPK장소" ON "DBP0103"."LOCATION" ("ROOMNUM", "LOCNUM")
-----
```

▲ LOCATION TABLE

```
-- DDL for Table MENTEE
-----
CREATE TABLE "DBP0103"."MENTEE"
(
  "TNUM" NUMBER(4,0),
  "MNUM" NUMBER(4,0)
)
-----
-- DDL for Index XPK소규모팀_기록
-----

CREATE UNIQUE INDEX "DBP0103"."XPK소규모팀_기록" ON "DBP0103"."MENTEE" ("TNUM", "MNUM")
-----
```

▲ MENTEE TABLE

06. 데이터베이스 설계

테이블 스키마

-- DDL for Table MENTOR

```
CREATE TABLE "DBP0103"."MENTOR"  
(  
  "MENTORNUM" NUMBER(4,0),  
  "PROFILE" VARCHAR2(400),  
  "MNUM" NUMBER(4,0),  
  "FIELD" NUMBER(4,0),  
  "STAR" FLOAT(126)  
)
```

-- DDL for Index XPK멘토

```
CREATE UNIQUE INDEX "DBP0103"."XPK멘토" ON "DBP0103"."MENTOR" ("MENTORNUM")
```

▲ MENTOR TABLE

-- DDL for Table REPORT

```
CREATE TABLE "DBP0103"."REPORT"  
(  
  "RPTNUM" NUMBER(2,0),  
  "TNUM" NUMBER(4,0),  
  "URL" VARCHAR2(200),  
  "RPTUPDATE" DATE,  
  "URL_RCP" VARCHAR2(200)  
)
```

-- DDL for Index XPK보고서

```
CREATE UNIQUE INDEX "DBP0103"."XPK보고서" ON "DBP0103"."REPORT" ("RPTNUM")
```

▲ REPORT TABLE

-- DDL for Table RENT_HIS

```
CREATE TABLE "DBP0103"."RENT_HIS"  
(  
  "ROOMNUM" NUMBER(2,0),  
  "LOCNUM" NUMBER(4,0),  
  "TNUM" NUMBER(4,0),  
  "RDATE" DATE,  
  "RTIME" CHAR(5),  
  "RNUM" NUMBER(4,0)  
)
```

-- DDL for Index XPK장소대여내역

```
CREATE UNIQUE INDEX "DBP0103"."XPK장소대여내역" ON "DBP0103"."RENT_HIS" ("RNUM")
```

▲ RENT_HIS TABLE

-- DDL for Table REVIEW

```
CREATE TABLE "DBP0103"."REVIEW"  
(  
  "REVNUM" NUMBER(4,0),  
  "MNUM" NUMBER(4,0),  
  "MENTORNUM" NUMBER(4,0),  
  "REVTITLE" VARCHAR2(30),  
  "REVCONTENT" VARCHAR2(400),  
  "REVDATE" DATE,  
  "CNT" NUMBER(4,0)  
)
```

-- DDL for Index XPK후기_게시글

```
CREATE UNIQUE INDEX "DBP0103"."XPK후기_게시글" ON "DBP0103"."REVIEW" ("REVNUM")
```

▲ REVIEW TABLE

06. 데이터베이스 설계

테이블 스키마

-- DDL for Table STAR

```
CREATE TABLE "DBP0103"."STAR"
(
  "MENTORNUM" NUMBER(4,0),
  "SATISFACTION" NUMBER(1,0),
  "PARTICIPATION" NUMBER(1,0),
  "PREPARATION" NUMBER(1,0),
  "STARTTOTAL" NUMBER(2,1),
  "REVNUM" NUMBER(4,0)
)
```

-- DDL for Index XPK별점

```
CREATE UNIQUE INDEX "DBP0103"."XPK별점" ON "DBP0103"."STAR" ("REVNUM")
```

▲ STAR TABLE

-- DDL for Table TEAM_LARGE

```
CREATE TABLE "DBP0103"."TEAM_LARGE"
(
  "TNUM" NUMBER(4,0),
  "MENTORNUM" NUMBER(4,0),
  "START_DATE" DATE,
  "FIELD" NUMBER(4,0),
  "MEETING_LOC" VARCHAR2(120),
  "TNAME" VARCHAR2(300),
  "LIMIT" NUMBER(3,0)
)
```

-- DDL for Index XPK대규모

```
CREATE UNIQUE INDEX "DBP0103"."XPK대규모" ON "DBP0103"."TEAM_LARGE" ("TNUM")
```

▲ TEAM_LARGE TABLE

-- DDL for Table TEAM_LARGE_HIS

```
CREATE TABLE "DBP0103"."TEAM_LARGE_HIS"
(
  "MNUM" NUMBER(4,0),
  "TNUM" NUMBER(4,0)
)
```

-- DDL for Index XPK대규모팀_기록

```
CREATE UNIQUE INDEX "DBP0103"."XPK대규모팀_기록" ON "DBP0103"."TEAM_LARGE_HIS" ("MNUM", "TNUM")
```

▲ TEAM_LARGE_HIS TABLE

-- DDL for Table TEAM_SMALL

```
CREATE TABLE "DBP0103"."TEAM_SMALL"
(
  "TNUM" NUMBER(4,0),
  "TNAME" VARCHAR2(300),
  "START_DATE" DATE,
  "FIELD" NUMBER(4,0),
  "MEETING_LOC" VARCHAR2(120),
  "END_DATE" DATE,
  "SUPPORT" NUMBER(7,0),
  "MENTORNUM" NUMBER(4,0)
)
```

-- DDL for Index XPK소규모

```
CREATE UNIQUE INDEX "DBP0103"."XPK소규모" ON "DBP0103"."TEAM_SMALL" ("TNUM")
```

-- DDL for Index TEAM_SMALL_UK1

```
CREATE UNIQUE INDEX "DBP0103"."TEAM_SMALL_UK1" ON "DBP0103"."TEAM_SMALL" ("TNAME")
```

▲ TEAM_SMALL TABLE

06. 데이터베이스 설계

테이블 스키마

```
-- DDL for Table WAITING
```

```
CREATE TABLE "DBP0103"."WAITING"  
(  
  "WAITNUM" NUMBER(4,0),  
  "MNAME" VARCHAR2(20),  
  "MNUM" NUMBER(4,0),  
  "TNUM" NUMBER(4,0)  
)
```

```
-- DDL for Index WAITING_PK
```

```
CREATE UNIQUE INDEX "DBP0103"."WAITING_PK" ON "DBP0103"."WAITING" ("WAITNUM", "TNUM")
```

▲ WAITING TABLE

07

클래스 설계

07. 클래스 설계

Domain Class

- ▼ model
 - > Field.java
 - > Location.java
 - > Member.java
 - > Mentee.java
 - > Mentor.java
 - > RentHis.java
 - > Report.java
 - > Review.java
 - > Star.java
 - > TeamLarge.java
 - > TeamLargeHis.java
 - > TeamSmall.java
 - > Waiting.java

- ① Field-> 분야(IT,BUSINESS 등) – fieldnum,fieldname
- ② Location-> roomnum,locnum,locname,address,city
- ③ Member -> membernum(pk), id ,pwd, name, sex, nickname, tel, birth, email, field
- ④ Mentee -> tnum, mnum, Member member, TeamSmall teamSmall
- ⑤ Mentor -> tnum,mnum
- ⑥ RentHis -> 대여기록 – roomnum,locnum,tnum,radate,rtime,rnum
- ⑦ Report -> 보고서 – rptnum,tnum,url,rptUpdate,url_rcp(영수증),expense(사용금액)
- ⑧ Review -> 후기 – revnum,revtitle,revcontent,revdate,cnt(조회수),star(별점)
- ⑨ Star -> 별점(멘토 추천 시 사용)
- ⑩ TeamLarge -> num, name, mentorNum, startDate, field1, meetingLoc, limit(최대 인원 수)
- ⑪ TeamLargeHis -> 특강기록 – teamNum, memberNum
- ⑫ TeamSmall -> num, name, mentorNum, startDate, field1, meetingLoc, endDate, support(지원금), Mentee mentee
- ⑬ Waiting -> 대기자명단 – waitnum, mname, mnum, tnum

07. 클래스 설계

DAO

model.dao

- > ConnectionManager.java
- > FieldDAO.java
- > JDBCUtil.java
- > LocationDAO.java
- > MemberDAO.java
- > MenteeDAO.java
- > MentorDAO.java
- > RentHisDAO.java
- > ReportDAO.java
- > ReviewDAO.java
- > StarDAO.java
- > TeamLargeDAO.java
- > TeamLargeHisDAO.java
- > TeamSmallDAO.java
- > WaitingDAO.java

- ① ConnectionManager → DataSource 생성/설정, DBP0103 연결
- ② FieldDAO → fieldNum(pk)로 필드/필드리스트 검색
- ③ JDBCUtil → JDBC API 사용 코드의 중복성과 복잡성을 제거하기 위한 utility class
- ④ LocationDAO → findLocationByXXX → 주로 location을 검색하는 기능
- ⑤ MemberDAO → 멤버 생성, 수정, 검색, 멤버 존재 유무 확인, 관심분야로 멘토 추천
- ⑥ MenteeDAO → 멘티 생성, 삭제, 멘티 리스트 검색
- ⑦ MentorDAO → 멘토 생성, 수정, 검색, 멘토 리스트 검색, 활동분야별 멘토 리스트
- ⑧ RentHisDAO → 장소대여기록 생성, 삭제, locname을 통한 대여기록 검색 등 검색
- ⑨ ReportDAO → 보고서등록, 수정, 삭제, rptnum을 통해 검색
- ⑩ ReviewDAO → 리뷰글등록, 수정, 삭제, revnum을 통해 리뷰글검색, 조회수증가, 별점설정을 위한 메소드
- ⑪ StarDAO → 멘토 추천 시 이용 / 별점 생성, 삭제, 별점합산, 멘토의 별점 검색
- ⑫ TeamLargeDAO → 팀 large 개설, 수정, 삭제, tnum을 통해 검색, 멘티로 참여했던 기록 검색하기 위한 메소드, (활동)분야별 팀 large리스트 검색
- ⑬ TeamLargeHisDAO → 팀 large기록 추가, 삭제, 검색, 현재까지 팀에 참여한 인원수 구하는 메소드
- ⑭ TeamSmallDAO → 팀 small 개설, 수정, 삭제, tnum을 통해 검색, 팀 리스트 검색, 팀 존재 유무 확인, 지원금 업데이트
- ⑮ WatingDAO → 대기자 생성, 삭제, 대기자 리스트 검색

07. 클래스 설계

Manager

model.service

- > ExistingMemberException.java
- > ExistingMentorException.java
- > ExistingTeamException.java
- > FieldManager.java
- > LocationManager.java
- > MemberManager.java
- > MemberNotFoundException.java
- > MenteeManager.java
- > MentorManager.java
- > MentorNotFoundException.java
- > PasswordMismatchException.java
- > RentHisManager.java
- > ReportManager.java
- > ReviewManager.java
- > SameUserIdException.java
- > StarManager.java
- > TeamLargeHisManager.java
- > TeamLargeManager.java
- > TeamNotFoundException.java
- > TeamSmallManager.java
- > WaitingManager.java

- ① Existing...Exception -> 존재하는 것들에 대한 예외처리
- ② FieldManager -> field검색에 대한 fieldDAO호출(int형으로 저장되어있어서 int(fieldnum) -> string(fieldname)변환 필요)
- ③ LocationManager -> location검색을 위한 locationDAO호출
- ④ MemberManager -> 멤버생성, 검색, 로그인, 멘토추천해주는DAO호출
- ⑤ ...NotFoundException -> 존재하지않는 것들에 대한 예외처리
- ⑥ RentHisManager -> 대여기록관리(대여기록 추가, 삭제, 검색)를 위한 DAO 호출
- ⑦ ReportManager -> 보고서관리(보고서등록, 검색, 삭제)를 위한 DAO호출
- ⑧ ReviewManager -> 후기게시판관리(리뷰생성,검색)을 위한 DAO호출
- ⑨ StarManager -> 후기게시판/멘토 별점관리(별점생성,수정,삭제,별점 합하기)를 위한 DAO호출
- ⑩ TeamSmall/LargeManager -> 팀 개설,삭제,검색을 위한 DAO호출
- ⑪ WaitingManager -> 팀 가입 신청할 시 들어가게 되는 대기자 명단을 관리(대기자명단에 등록, 삭제, 검색)하는 DAO호출

07. 클래스 설계

Controller

controller.member

- > `ListStarFieldRecommendController.java` → 사용자가 관심분야로 선택한 field에 따라 멘토를 별점순으로 나열
- > `LoginController.java` → request parameter값 이용한 로그인 기능(세션 설정)
- > `LogoutController.java` → 세션에 저장된 사용자 아이디를 삭제하고 세션을 무효화 함
- > `MainPageController.java` → 메인페이지로 넘어감
- > `MemberSessionUtils.java` → 로그인 상태 확인
- > `MyPageController.java` → 로그인한 회원의 상세정보로 넘어감
- > `myPageTeamController.java` → 내가 참여한 팀들의 상세정보로 넘어감
- > `RegisterMemberController.java` → 멤버로 회원가입

06. 클래스 설계

Controller

controller.mentor

- > DetailMentorController.java → 멘토의 상세정보 출력 기능
- > JoinMentorController.java → 멘토로 등록하는 기능
- > ListMentorController.java → 멘토 리스트를 출력하는 기능
- > MentorSessionUtils.java → 멘토 세션 확인
- > ViewMentorListJsonController.java → Ajax로 리스트를 현재 화면에 포함하는 기능

06. 클래스 설계

Controller

controller.team

- > `AssentTeamSmallMentorController.java` → 멘토 권한으로 멘티 승인 기능
- > `DeleteTeamSmallController.java` → 팀 삭제 기능
- > `DetailRequestTeamSmallController.java` → 멘티가 팀에 요청한 뒤 상세정보 보기 페이지로 돌아오는 기능
- > `DetailTeamSmallController.java` → 팀 상세정보 기능
- > `JoinTeamSmallFormController.java` → 팀을 만드는 기능
- > `LeaveTeamController.java` → 멘티들이 팀을 탈퇴하는 기능
- > `ListTeamSmallController.java` → 팀 리스트 출력 기능
- > `ListWaitingController.java` → 팀의 대기자명단 출력 기능
- > `RequestTeamSmallController.java` → 팀에 가입 요청하는 기능
- > `TeamSmallListSessionUtils.java` → 팀 리스트 세션 관리
- > `TeamSmallSessionUtils.java` → 팀 세션 관리
- > `ViewTeamSmallJsonController.java` → Ajax로 선택한 기능에 따라 현재 보는 페이지에 리스트 포함하는 기능

06. 클래스 설계

Controller





controller.teamLarge

- > AttendTeamLargeController.java → 팀라지에 수강생으로 등록하는 기능
- > DetailAttendTeamLargeController.java → 수강생으로 등록한 뒤 세부사항으로 돌아오는 기능
- > DetailTeamLargeController.java → 팀라지의 세부사항으로 출력하는 기능
- > JoinTeamLargeController.java → 팀라지를 만드는 기능
- > ListTeamLargeController.java → 팀라지 리스트를 출력하는 기능
- > TeamLargeListSessionUtils.java → 팀라지 세션 관리
- > ViewTeamLargeJsonController.java → Ajax로 리스트를 현재 화면에 포함하는 기능

06. 클래스 설계

Controller

controller.review

- >  DetailReviewController.java → 리뷰 세부사항 기능
- >  JoinFormReviewController.java → 리뷰 등록 페이지에 자기가 속한 팀스몰, 팀라지의 멘토리스트를 포함시키는 기능
- >  JoinReviewController.java → 리뷰 등록하는 기능
- >  ListReviewController.java → 리뷰 리스트 기능

06. 클래스 설계

Controller

controller.rent

- > DetailRentPlaceController.java → 장소대여 세부사항 기능
- > ListRentController.java → 장소대여 리스트 출력 기능
- > RentListSessionUtils.java → 장소대여 세션 관리
- > RequestFormRentPlaceController.java → 자기가 멘토인 팀스몰 리트르를 장소대여폼에 출력하는 기능
- > RequestRentPlaceController.java → 장소대여 신청 기능

06. 클래스 설계

Controller

controller.report

> JoinReportController.java → 보고서 등록 기능

> ListReportController.java → 보고서 리스트 출력 기능

08

Request URI 정의 및 처리 흐름

08. Request URI 정의 및 처리 흐름

Request URI 정의

```
public void initMapping() {

    mappings.put("/", new ForwardController("index.jsp"));
    mappings.put("/member/login", new LoginController());
    mappings.put("/member/login/form", new ForwardController("/member/loginForm.jsp"));
    mappings.put("/member/logout", new LogoutController());
    mappings.put("/member/register", new RegisterMemberController());
    mappings.put("/member/register/form", new ForwardController("/member/memberJoin.jsp"));

    mappings.put("/main/mainpage", new MainPageController());
    mappings.put("/mypage/mypage", new MyPageController());

    mappings.put("/mentor/mentorList", new ListMentorController());
    mappings.put("/mentor/mentorJoin", new JoinMentorController());
    mappings.put("/mentor/mentorJoin/form", new ForwardController("/mentor/mentorJoin.jsp"));
    mappings.put("/mentor/mentorDetail", new DetailMentorController());
    mappings.put("/mentor/mentorListView/json", new ViewMentorListJsonController());

    mappings.put("/team/myTeamDetail", new myPageTeamController());
    mappings.put("/team/teamSmalljoin", new JoinTeamSmallFormController());
    mappings.put("/team/teamSmalljoin/form", new ForwardController("/team/teamSmallJoin.jsp"));
    mappings.put("/team/teamSmallList", new ListTeamSmallController());
    mappings.put("/team/teamSmallDetail", new DetailTeamSmallController());
    mappings.put("/team/teamSmallWaiting", new ListWaitingController());
    mappings.put("/team/teamSmallWaitinging", new ForwardController("/team/WaitingPage.jsp"));
    mappings.put("/team/teamSmalljoinMentee/form", new ForwardController("/team/teamSmallJoinMentee.jsp"));

    mappings.put("/team/teamLeave", new LeaveTeamController());
    mappings.put("/team/teamDelete", new DeleteTeamSmallController());

    mappings.put("/team/teamSmallRequest", new RequestTeamSmallController());
    mappings.put("/team/teamSmallWaiting/forming", new AssentTeamSmallMentorController());
    mappings.put("/team/teamSmallDetailRequest", new DetailRequestTeamSmallController());
    mappings.put("/team/teamSmallView/json", new ViewTeamSmallJsonController());
}
```

08. Request URI 정의 및 처리 흐름

Request URI 정의

```
mappings.put("/teamLarge/teamLargeList", new ListTeamLargeController());
mappings.put("/teamLarge/teamLargeDetail", new DetailTeamLargeController());
mappings.put("/teamLarge/teamLargeAttend/form", new ForwardController("/teamLarge/teamLargeAttend.jsp"));
mappings.put("/teamLarge/teamLargeAttend", new AttendTeamLargeController());
mappings.put("/teamLarge/teamLargeJoin", new JoinTeamLargeController());
mappings.put("/teamLarge/teamLargeJoin/form", new ForwardController("/teamLarge/teamLargeJoin.jsp"));
mappings.put("/teamLarge/teamLargeView/json", new ViewTeamLargeJsonController());
mappings.put("/teamLarge/teamLargeDetailAttend", new DetailAttendTeamLargeController());
```

```
mappings.put("/report/reportList", new ListReportController());
mappings.put("/report/reportJoin/form", new ForwardController("/report/reportJoin.jsp"));
mappings.put("/report/reportJoin", new JoinReportController());
```

```
mappings.put("/review/reviewList", new ListReviewController());
mappings.put("/review/reviewJoin", new JoinReviewController());
mappings.put("/review/reviewJoin/form", new JoinFormReviewController());
mappings.put("/review/reviewDetail", new DetailReviewController());
```

```
mappings.put("/rent/rentPlacePageList", new ListRentController());
mappings.put("/rent/rentPlacePageDetail", new DetailRentPlaceController());
mappings.put("/rent/rentPlaceRequest", new RequestRentPlaceController());
mappings.put("/rent/rentPlaceRequest/form", new RequestFormRentPlaceController());
```

```
logger.info("Initialized Request Mapping!");
```

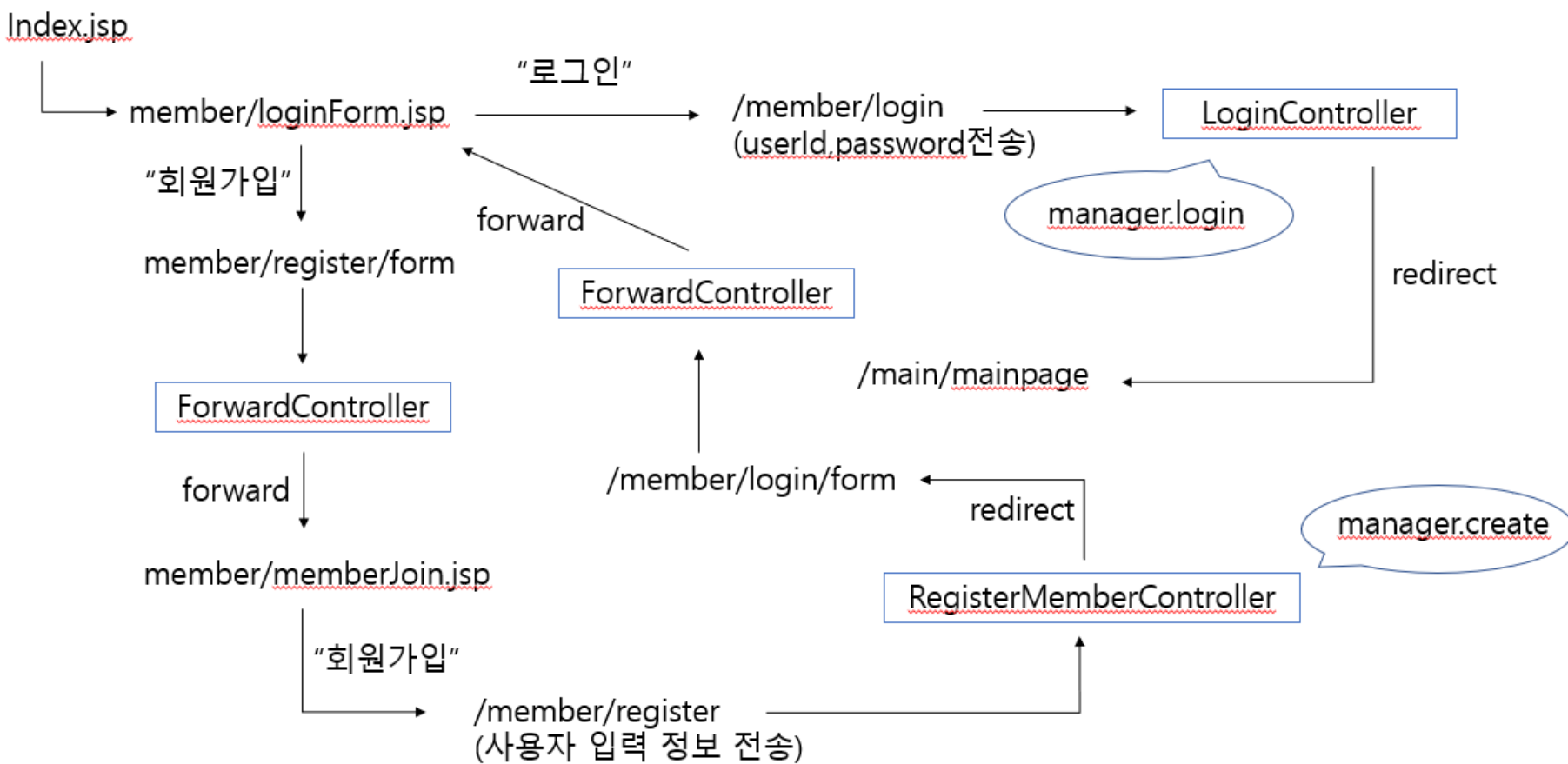
```
}
```

```
public Controller findController(String uri) {
    return mappings.get(uri);
}
```

```
}
```

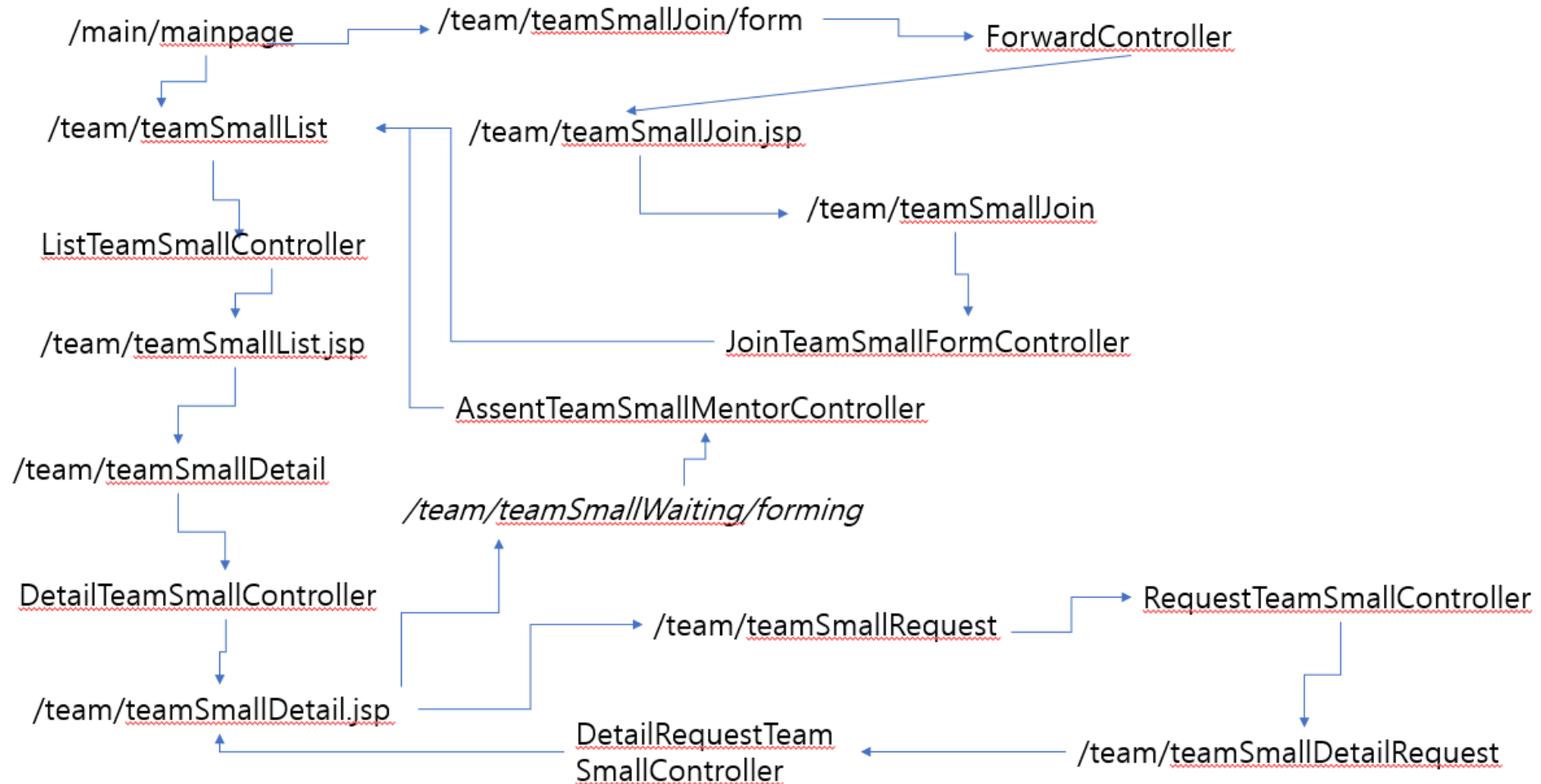
08. Request URI 정의 및 처리 흐름

처리 흐름-로그인 및 회원가입



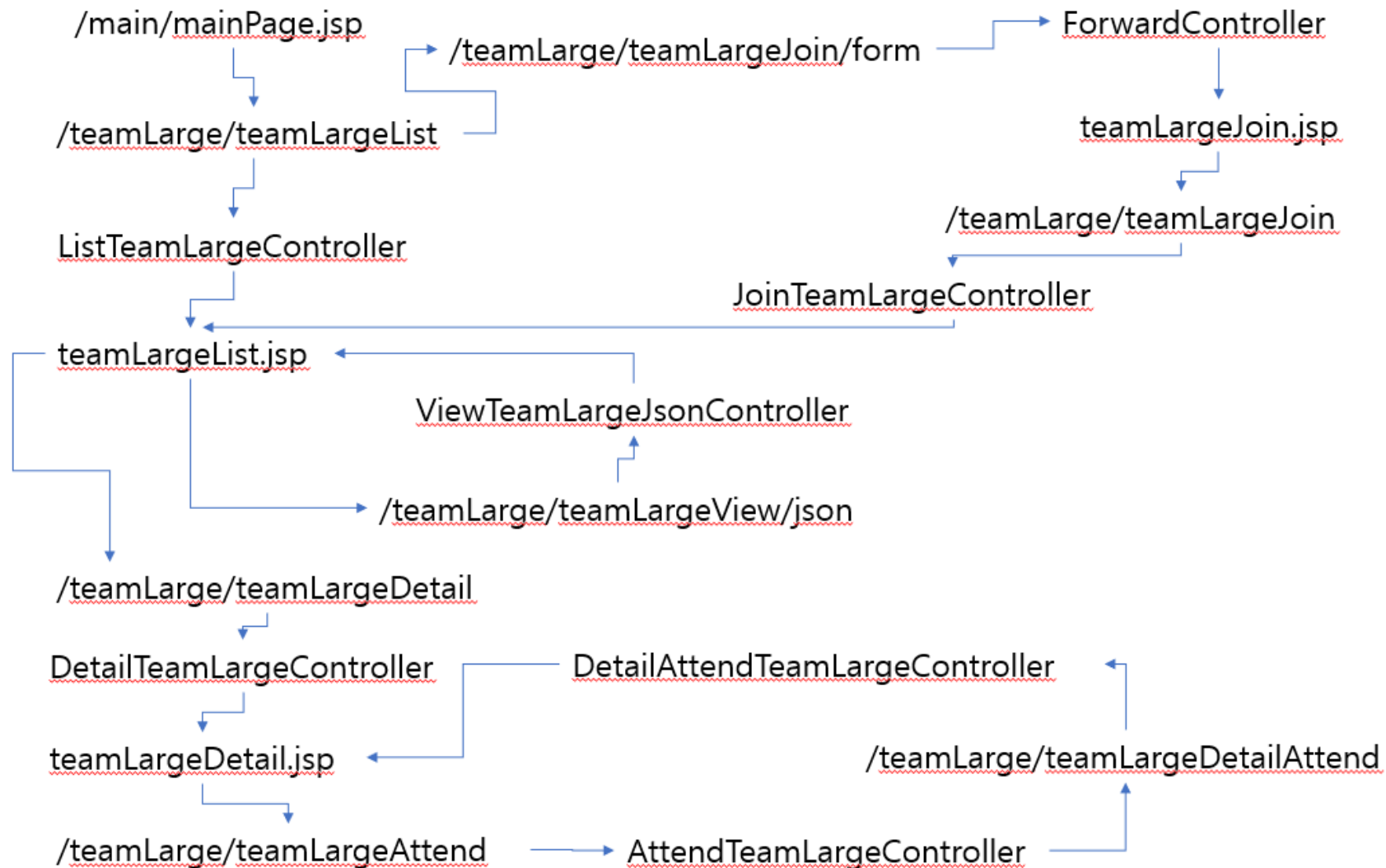
08. Request URI 정의 및 처리 흐름

처리 흐름 – 멘토링 팀(TeamSmall)



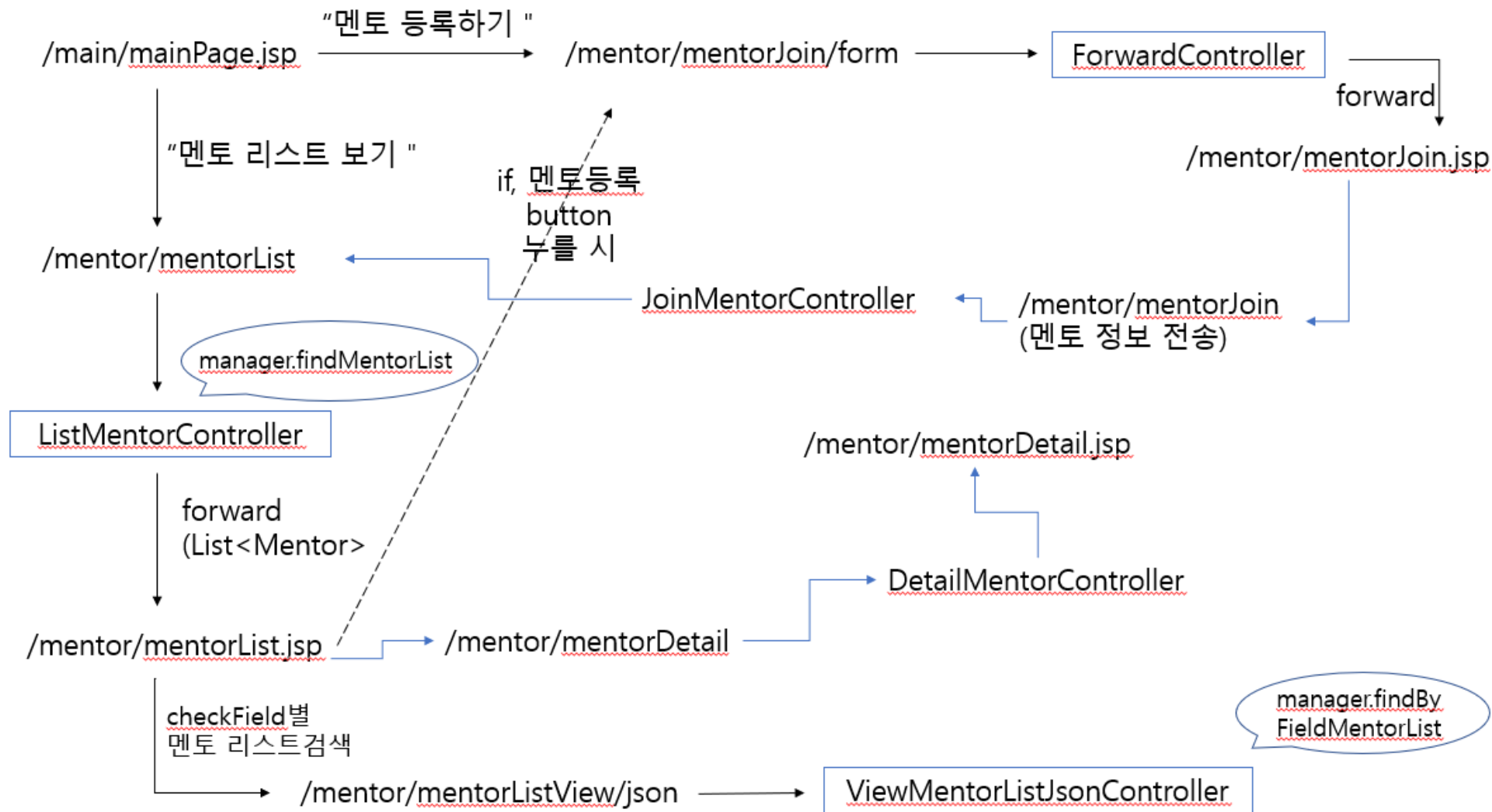
08. Request URI 정의 및 처리 흐름

처리 흐름 - 특강(TeamLarge)



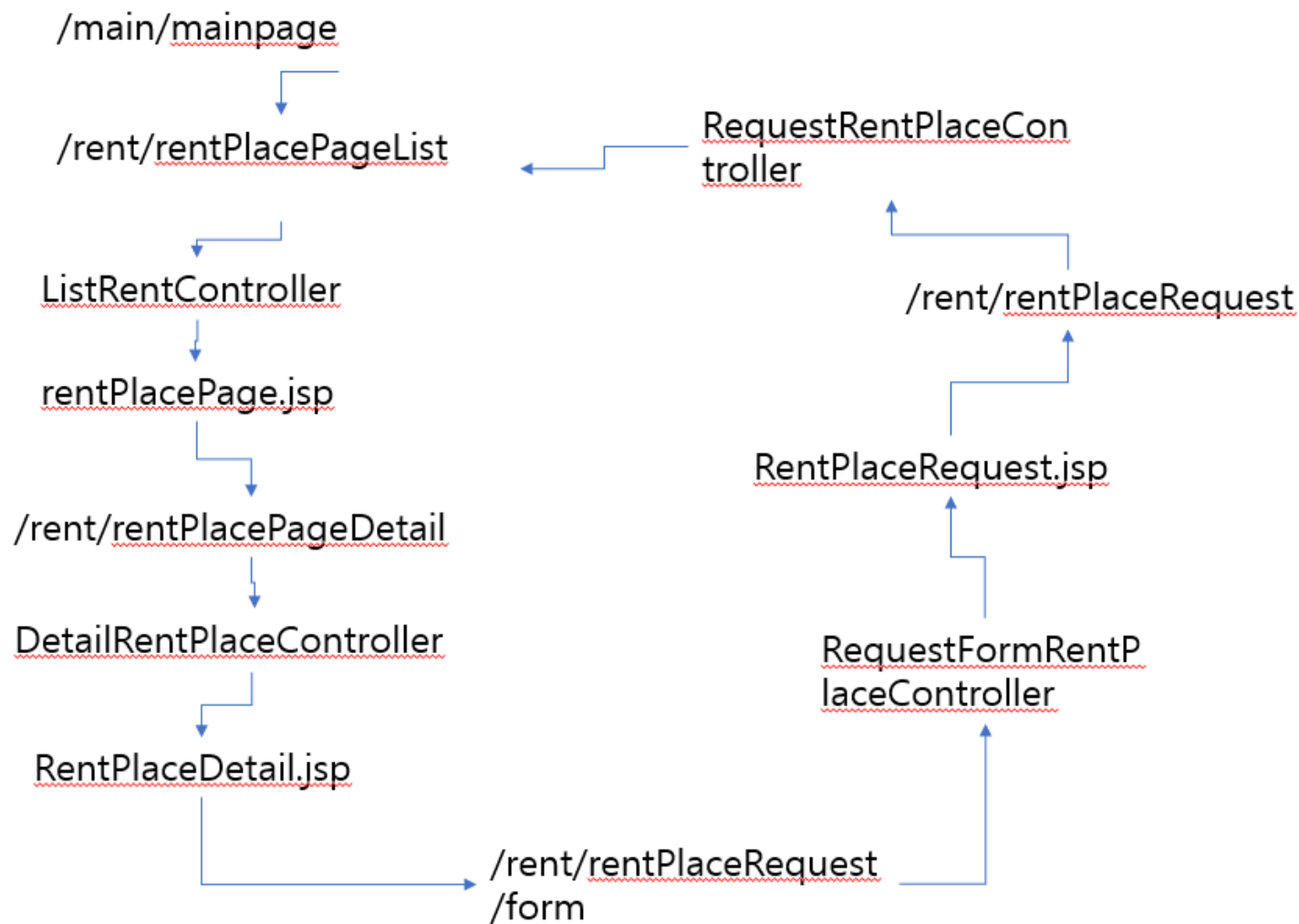
08. Request URI 정의 및 처리 흐름

처리 흐름 - 멘토



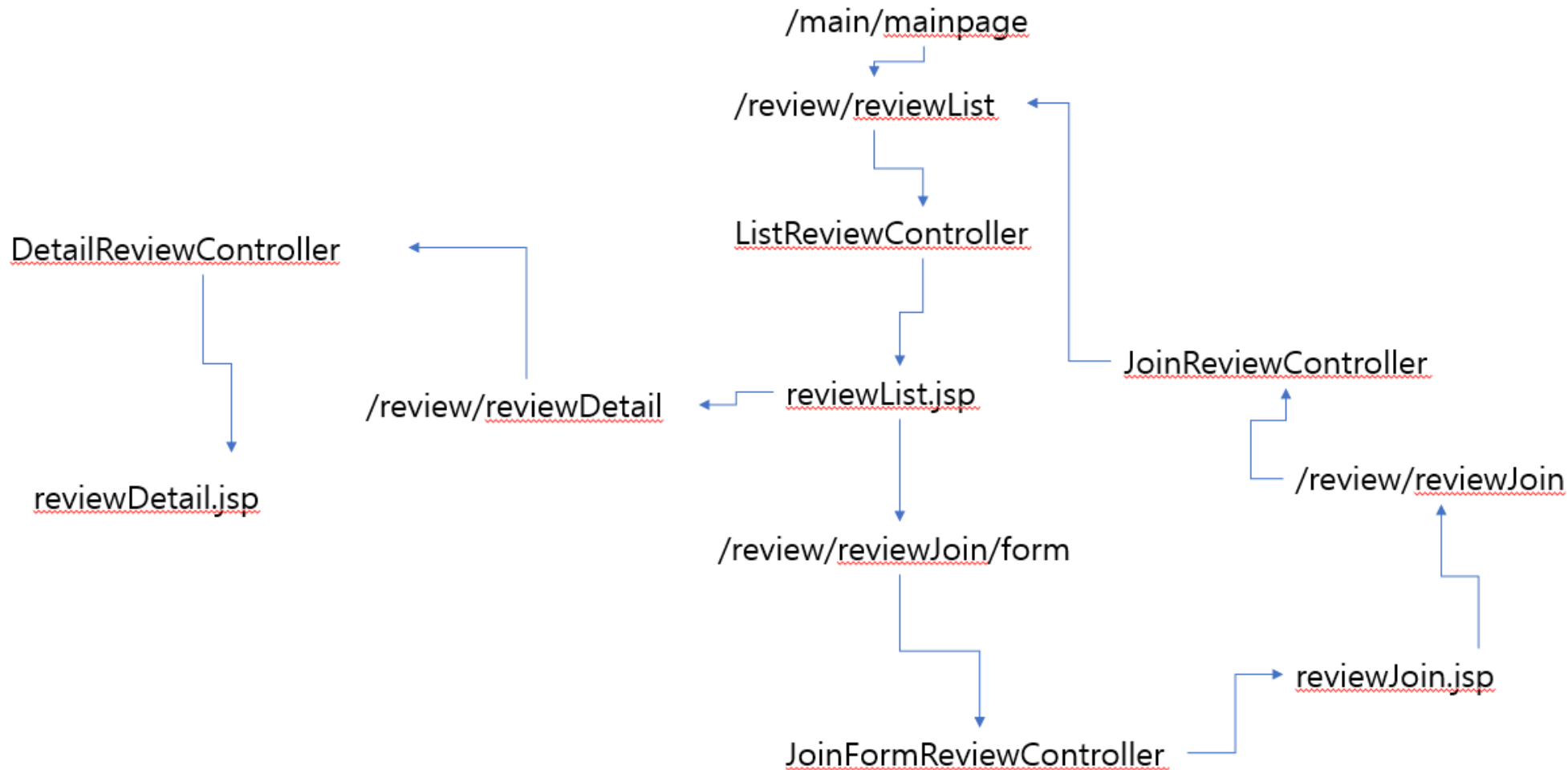
08. Request URI 정의 및 처리 흐름

처리 흐름 - 장소대여



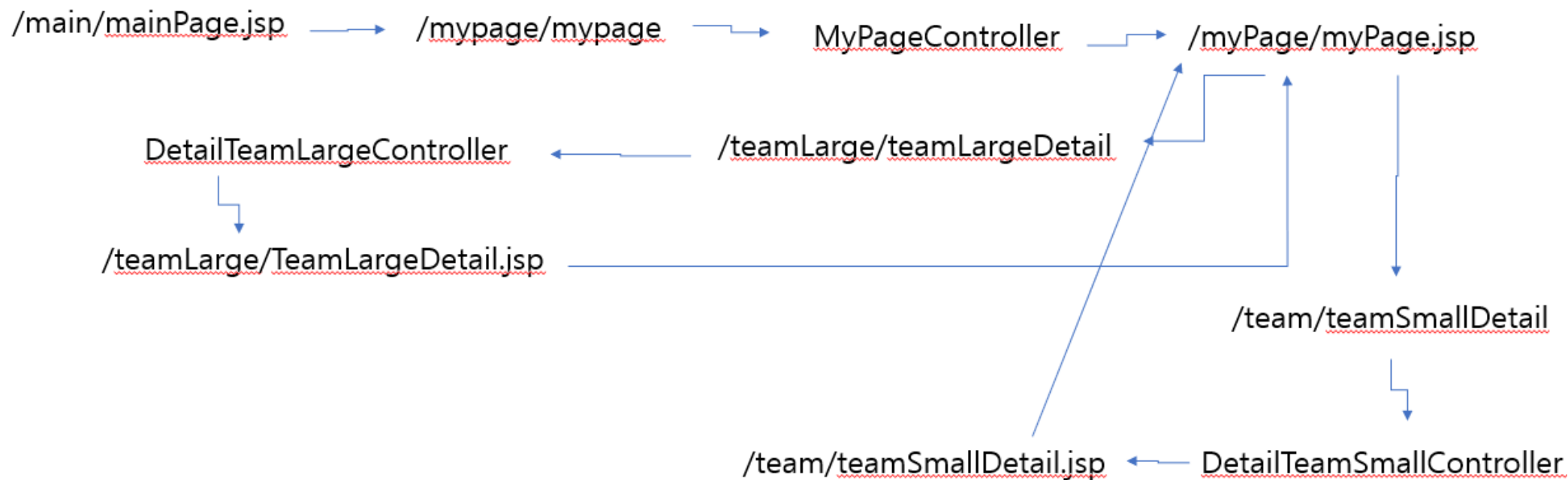
08. Request URI 정의 및 처리 흐름

처리 흐름 – 후기 게시판



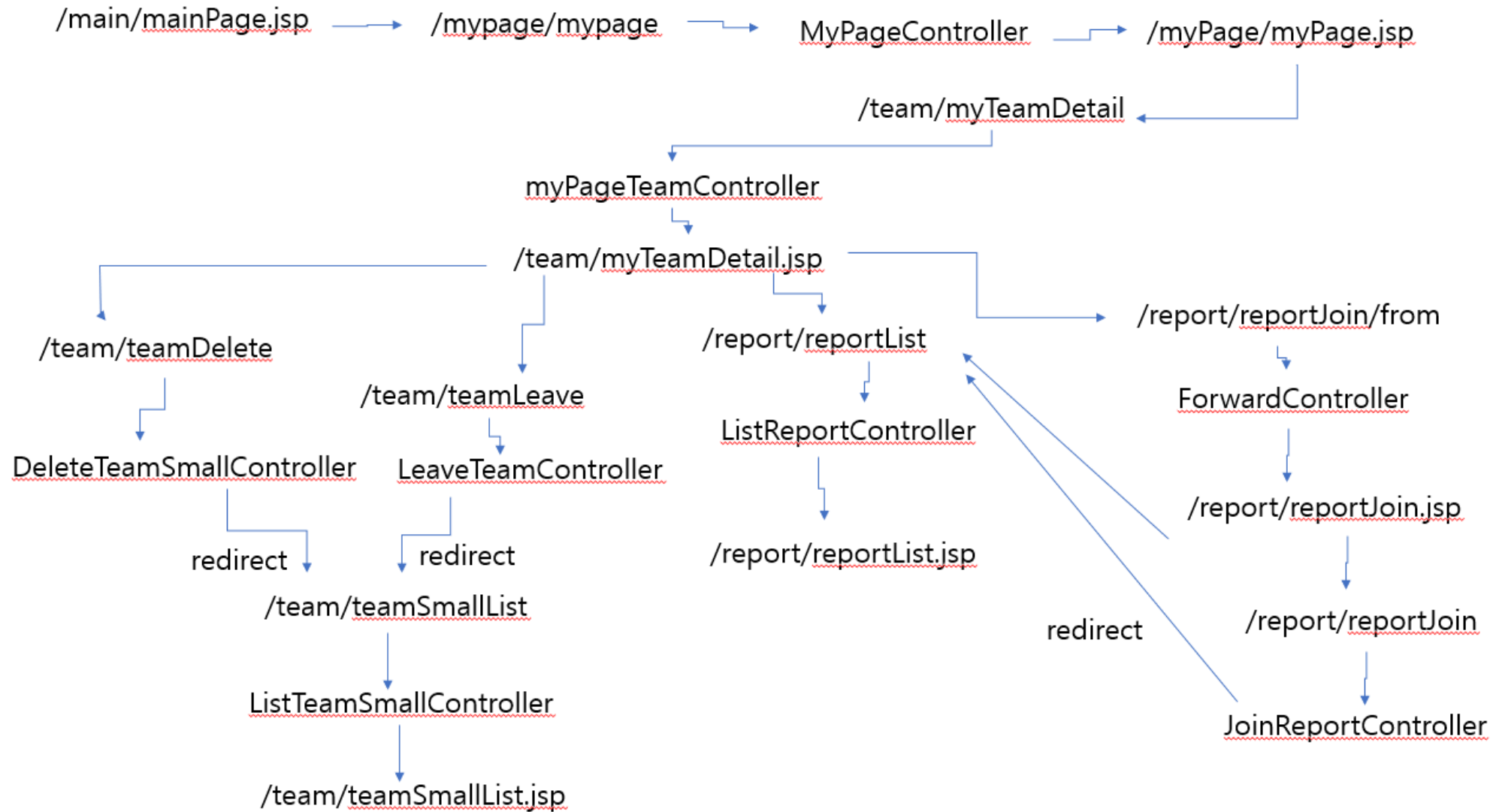
08. Request URI 정의 및 처리 흐름

처리 흐름 - 마이페이지



08. Request URI 정의 및 처리 흐름

처리 흐름 - 마이페이지

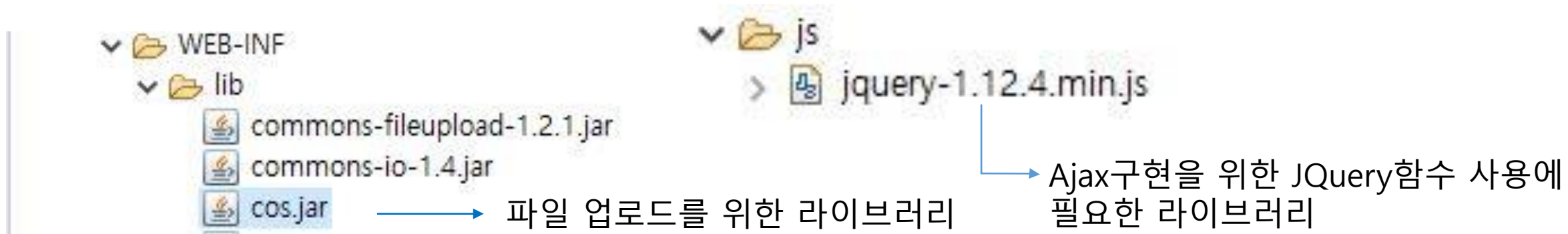


09

구현 방법

09 구현 방법

사용기술 및 라이브러리



- 개발 환경

개발IDE - Eclipse IDE for Java EE Developer

Tomcat - Apache Tomcat 9.0

Oracle - Oracle SQL Developer/Oracle Database 11g Express Edition

ERwin - ERwin Data Modeler(CA)

- 사용기술

Model-View-Controller구조

- * Model – JavaBeans로 데이터 캡슐화/핵심업무 로직의 구현(Business layer)
- * View – JSP,HTML로 프레젠테이션 책임짐(Presentation layer)
- * Controller – Servlet으로 client로부터 HTTP Request수신
SQL문을 사용, DAO클래스를 이용해 DB관리(Persistence layer)

09 구현 방법

알고리즘

<팀 small 구성 알고리즘>

- (1) 멘토가 팀을 개설한다.
- (2) 회원들은 원하는 팀에 가입을 요청한다.
- (3) 멘토가 요청에 대해 승인/거절한다.
- (4) 승인할 시 회원은 그 팀의 멘티가 된다.
- (5) 멘티가 4명이 되면 그 팀은 모집완료가 되며 다른 회원은 이 팀에 가입요청을 할 수 없다.

<팀 small 지원금 지급 알고리즘>

- (1) 팀small은 멘토링 활동에 대한 보고서와 영수증사진 파일을 업로드한다.
- (2) 업로드할 시 사용금액을 입력한다.
- (3) 첫 등록하는 보고서일 경우 멘토링활동시작으로 간주하고 50만원을 지급한다.
- (4) 남은 지원금은 팀 상세보기에서 확인 가능하다.
- (5) 올린 파일은 보고서 리스트에서 확인 가능하다.

09 구현 방법

알고리즘

<팀 Large 구성 알고리즘>

- (1) 멘토가 팀을 개설한다. 멘토가 아닐 시 멘토 등록 페이지로 넘어간다.
- (2) 멘토가 팀을 개설할 때 제한 인원 수를 입력한다.
- (3) 회원들은 리스트에서 모집중인 팀 조회를 할 수 있고, 멘토가 설정한 인원이 다 안 찼을 시 가입 신청할 수 있다.

<별점 알고리즘>

- (1) 멘티들은 후기게시판에 자신의 멘토에 대한 후기글을 작성할 수 있다.
- (2) 해당 멘토에 대해 만족성, 참여성, 준비성 평가를 별점(1점 ~ 5점)으로 등록한다.
- (3) 이 3개의 항목에 대한 평균을 구해 리뷰 상 세보기 페이지에서 멘토의 총 별점을 확인할 수 있다.

09 구현 방법

알고리즘

<멘토 추천 알고리즘>

- (1) 회원은 회원가입할 때 관심분야 3가지를 선택한다.
- (2) 앞서 작성한 별점 알고리즘으로 멘토의 별점이 확정된다.
- (3) 이 별점을 기반으로 회원이 선택한 분야 각각의 멘토들을 추천받을 수 있다.
목록에는 멘토이름(ID), 별점, 멘토로 활동했던 경험 횟수가 표시된다. rownum
을 이용해 별점 상위 5명인 멘토들만 목록에 표시하도록 했다.

09 구현 방법

문제해결 방법

- java.sql.SQLIntegrityConstraintViolationException: ORA-01400: NULL을 ("DBP0103"."TEAM_SMALL"."TNAME") 안에 삽입할 수 없습니다

해결방법 : request.getParameter의 변수명 문제

- 심각: Servlet.service() for servlet [jsp] threw exception
java.lang.NullPointerException

at org.apache.jsp.mentor.mentorJoin_jsp._jspService(mentorJoin_jsp.java:258)

해결방법 : Request Mapping이 잘 이루어졌는지 확인

- String과 Date 사이의 형변환

해결방법 : java.util.Date, java.sql.Date, java.text.SimpleDateFormat API 문서 참조

09 구현 방법

문제해결 방법

- `SQLException` 부적합한 열 이름

해결방법 : 컬럼에 대한 이름이 DB와 같게 되어있는지 확인

- `java.io.IOException: Posted content type isn't multipart/form-data`

해결방법 : form 속성을 post로하고 enctype="multipart/form-data"를 추가한다

- `java.sql.SQLIntegrityConstraintViolationException: ORA-02292: 무결성 제약조건(DBP0103.R_56)이 위배되었습니다- 자식 레코드가 발견되었습니다`

해결방법 : 무결성제약조건이 위배되었다. TNUM을 외래키로 사용하는 RentHis테이블이 있는데 팀 삭제를 하면서 renthis도 같이 삭제하지 않아 생긴 문제였다.

결론 및 소감

DOUM

이슬기 : 실제 운영할 수 있는 홈페이지를 만드는데 있어 많은 시간과 노력이 들어간다는걸 깨달았다. 효율적인 테이블을 만들기 위해서 논리적 모델링부터 물리적 모델링까지 한 달에 가까운 시간동안 차근차근 단계를 밟아 갔다. MVC구조에 기반한 시스템 설계를 위해 Controller, Model, View로 나누고 요청 처리 흐름을 정의했고, 초반에 생각했던 기능들을 모두 구현했다. Controller를 사용하면서 Manager와 dao 없이는 할 수 있는게 없다는걸 깨달았고, dao의 sql문에서 미세한 차이에 따라 쓸 수 없는 요소들이 많아 후반에 가서는 직접 dao를 구현하게 되었다. 3명이서 하게 되어 벅차다고 느꼈지만 이렇게 잘 마무리할 수 있게 되어 뿌듯하다.

김예진 : 한학기 동안 MVC 구조의 전반적인 지식을 다루고 배우는 것에 그치지 않고 직접 팀원들과 함께 성과물을 이루어 내어 유익했다. 처음에는 이 기능들을 다 구현할 수 있을지 걱정도 되고 우리 생활에서도 흔히 쓰이는 웹페이지를 내가 만든다는 사실이 신기하기만 했는데 간단한 동작을 하는것도 매우 많은 노력이 들어간다는 것을 알고 간단한 웹페이지를 쓰더라도 고마움을 느끼며 쓰게 될 것이다. 또한 팀 프로젝트를 장기간에 걸쳐 한 경험은 이번이 처음인데 업무 분담이 얼마나 중요하며 MVC구조가 업무분담에 매우 유용한 것을 알게 되었다.

이예건 : 살면서 이용했던 많은 사이트들의 개발자들의 마음을 이해해보는 시간을 가졌다. 클릭만으로 편하게 이용했던 사이트들을 막상 실제로 구현해보니 클릭 하나에도 많은 생각이 필요하고 데이터값들에 대한 이동 또한 다루기 어렵다는 것을 알았다. 하나의 웹사이트를 만드는데 MVC패턴을 적용해 작업을 분담해서 할 수 있다는 것 또한 편했다. 이렇게 분담을 할 경우에는 데이터들에 대한 변수명, 파라미터값들 충분히 상의 후 통일해야한다는 조언도 얻었다. 처음엔 4명으로 시작했지만 한명의 휴학으로 인해 3명으로 진행하면서 시간도 많이 부족했고 한 사람이 해야할 일도 더 늘고 정말 힘들었지만 처음에 구현하려고했던 기능들도 모두 구현하고 오류없이 결과물을 내게 되어서 힘든 만큼 값진 경험이었다고 느꼈다.

업무 분장

DOUM

이슬기 : JSP, Controller, Request URI 정의 및 구현

김예진 : JSP, Manager, DAO 구현, 시스템 구성도(MVC 구조), 데이터베이스 설계

이예건 : JSP, Manager, DAO 구현, 요구분석 및 USE Case 구현, 데이터베이스 설계, 알고리즘