

프로젝트 수행 계획서

☐ 요약

연구 주제명	팀 자료관리 사이트 제작		
연구 내용 요약	<ul style="list-style-type: none"> • 유저와 서버 간 통신을 이용해 자료의 업로드/다운로드 서비스를 구현한다 • 그룹별 권한기능을 구현하여 사용자마다 다른 화면을 제공하도록 구현한다. • 자료 특성에 따른 효율적인 분류방법에 대해 고민한다. 		
팀 명	MBTI		
팀명 설명	<p>MBTI (Make Best Team Idea)</p> <p>최고의 팀을 만드는 생각</p> <p>최고의 팀을 만들기 위한 팀의 의지를 표현하고 있습니다.</p>		
팀 원	이 름		역 할
	팀 장	장민관	문서작업, 기능설계/구현
	팀 원 1	권진성	기능설계/구현
	팀 원 2	최정규	기능설계/구현
	팀 원 3	박주형	기능설계/구현
요구 기술 (확보 자료)	HTML, CSS, JavaScript, DB, JSP		
기대효과	<ul style="list-style-type: none"> • 팀의 자료관리 사이트를 제공한다. • 효율적인 자료분류 방법을 얻는다. • 사용자의 권한에 맞춘 필요 자료를 제공한다. 		

□ 세부 내용

1. 프로젝트 목표

1-1. 과제 목표

- 웹(HTML,CSS,JavaScript)과, 서버를 활용하여 프론트엔드와 백엔드 기술을 프로젝트에 적용한다.
- 효율적인 자료관리 방법을 찾아 구현한다.
- 그룹별 권한을 통해 사용자별 다른 환경을 제공한다.

1-3. 최종 결과물

- 팀 자료관리 사이트
- 문서화된 산출물(계획서, 요구분석서, UI 설계서 등)
- 결과 보고서(문서 및 ppt)
- 발표 동영상(소개 및 시연)

2. 추진내용

2-1. 추진내용(전체 내용 정리)

■ 자료관리

- ▶ 파일 업로드/수정/삭제 기능
- ▶ 중요파일 설정기능 (중요도 설정)

■ 자료분류

- ▶ 프로젝트별/파일 이름별/종류별/중요도 분류

■ 권한관리

- ▶ 그룹별/개인별/전체 파일 접근권한 관리 (읽기/쓰기/삭제권한)

■ 그룹관리

- ▶ 사용자 검색 기능
 - 사용자 고유번호(학번, 사번)를 이용한 검색
- ▶ 사용자 정보 확인
 - 연락처/이메일/기타정보 확인기능
- ▶ 그룹설정
 - 그룹 추가/수정/제거 기능

■ 로그인/회원가입 기능

■ 팀(그룹) 소통기능

- ▶ 그룹 간 소통기능

■ 보조 기능

- ▶ 팀(그룹) 온 오프라인 확인기능
- ▶ 회의장소 탐색기능
 - 회의장소 (주변카페, 회의실 안내)
- ▶ 전체/그룹 공지기능

2-2. 추진 범위(실제 추진 내용만)

■ 자료관리

- ▶ 파일 업로드/수정/삭제 기능
- ▶ 중요파일 설정기능 (중요도 설정)

■ 자료분류

- ▶ 프로젝트별/파일 이름별/종류별/중요도 분류

■ 권한관리

- ▶ 관리자 계정/사용자 계정 분리
- ▶ 그룹별/개인별/전체 파일 접근권한 관리 (읽기/쓰기/삭제권한)

■ 그룹관리

- ▶ 사용자 검색 기능
 - 사용자 고유번호(학번, 사번)를 이용한 검색
- ▶ 그룹설정
 - 그룹 추가/수정/제거 기능

3. 기술기획의 추진전략 및 방법

3-1. 추진체계 구성

□ 팀원 간 역할 분담 계획

- 회의를 통해 나온 기능을 각 팀원별 맡아 구현 후 기능 병합과정을 통해 하나의 웹 사이트로 만든다.

3-2. 추진일정

일련 번호	기술기획내용	추진 일정 (월)														
		2022년 9월				2022년 10월				2022년 11월				2022년 12월		
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	요구분석															
2	자료수집															
3	기능 설계															
4	기능구현 및 병합															
5	테스트 및 수정															
6	발표 준비															

4. 기대성과

☐ 학습 측면의 효과

- JSP, 서블릿, DB 등을 활용하여 웹 프로젝트를 경험한다.
- 자료의 관리 방법을 생각하여 효율적으로 관리하는 방법을 찾는다.
- 그룹을 체계적으로 관리하는 방법을 찾는다.
- 팀 프로젝트를 통해 협업 능력을 키운다.
- 서버를 활용하는 방법을 익힌다.

☐ 개발한 프로그램(결과물)이 제공해 주게 될 효과

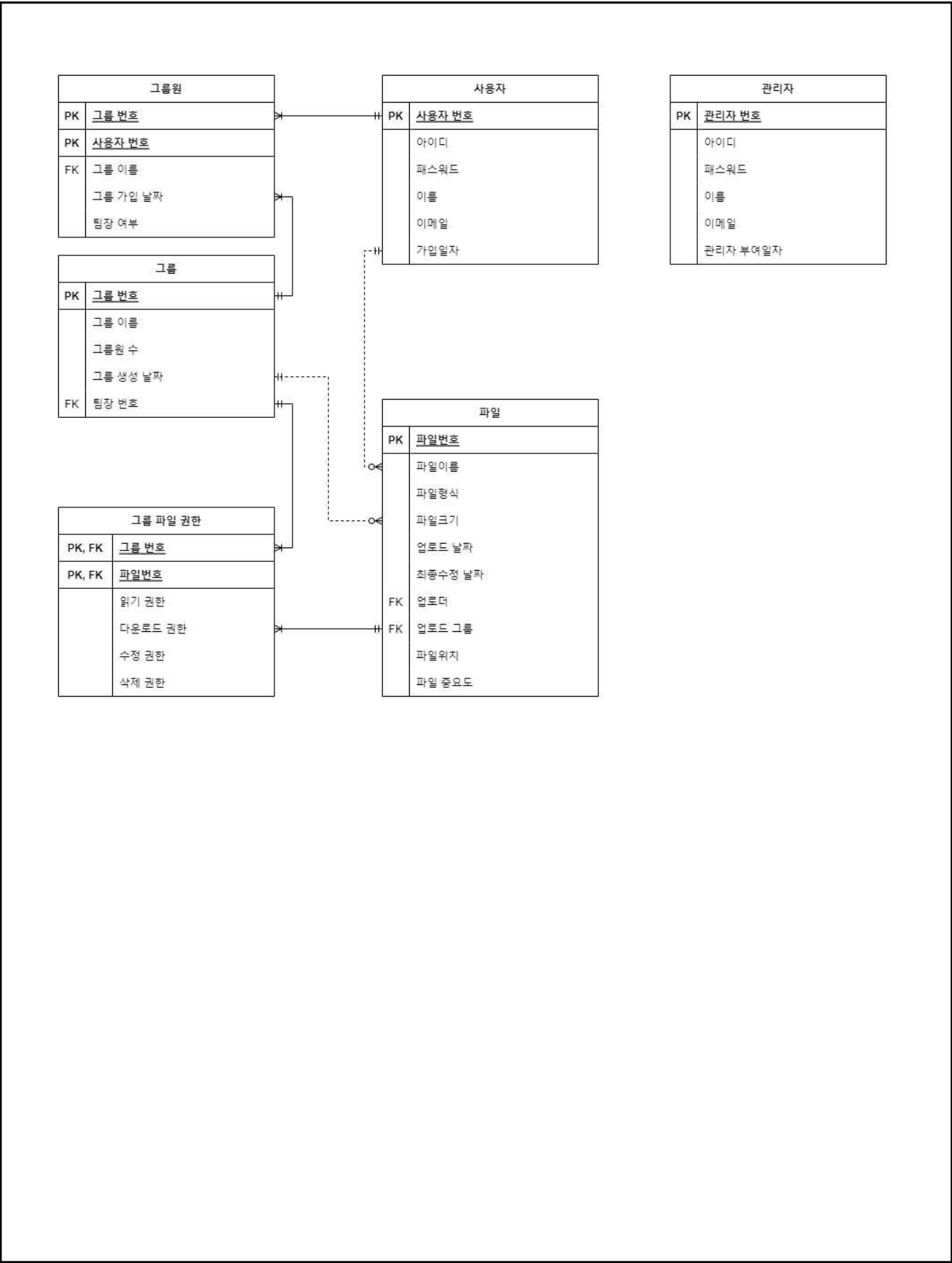
- 팀 자료 관리를 통해 효율적인 업무(프로젝트)환경을 제공한다.
- 그룹 권한별 파일 접근을 통해 불필요한 자료 노출을 줄인다.
- 자료의 분류를 통해 이용자가 원하는 자료를 빠르게 찾을 수 있다.

DB설계도

과목명	JSP와 서블릿
연구주제명	팀 자료관리 사이트
팀 명	B팀 (MBTI조)
팀 원	장민관, 권진성, 박주형, 최정규

2022. 11. 13.

제 목 : 실체 관계도(ERD)	단 계 : 업무 분석 및 설계	작성자 : B팀
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 1/1	승 인 : (인)



제 목 : 테이블 목록	단 계 : 업무 분석 및 설계	작성자 : B팀
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 1/1	승 인 : (인)

구 분	자 료 명	자 료 설 명(제약조건, 도메인 포함)
사용자	USER	사용자 테이블(사용자 정보)
관리자	ADMIN	관리자 테이블(관리자 정보)
그룹	GROUP	그룹 테이블(그룹 정보)
파일	FILE	파일 테이블(파일 정보)
그룹원	GROUP_MEMB	그룹원 테이블(그룹원 정보)
그룹 파일 권한	GROUP_PERMISSION	그룹 파일 권한 테이블(그룹별 파일 권한 정보)

Version 1.0

- ※ 작성 방법
- 구분란은 프로세스명, 데이터명, 코드명, 기타 용어 등으로 구분하여 서술함
 - 용어 설명 위주로 작성하여도 무방함

제 목 : 데이터 요소(테이블) 명세서	단 계 : 업무 분석 및 설계	작성자 :
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 1/6	승 인 : (인)

번호	엔티티명 (또는 파일명)	한글 : 사용자 영문 : User		설 명	특 성		
	한 글 명	영 문 명	필 드 명		Type	Length	Key
1	사용자 번호	USER NUMBER	U_NUM	사용자 고유번호	int	—	P.K
2	아이디	ID	U_ID	사용자 아이디	char	20	
3	패스워드	PASSWORD	U_PW	사용자 패스워드	char	20	
4	이름	NAME	U_NAME	사용자 이름	char	20	
5	이메일	E MAIL	U_EMAIL	사용자 이메일	char	50	
6	가입일자	JOIN DATE	U_JOIN_DATE	사용자 가입일자	char	8	

Version 1.0

※ 작성 방법 : 파일을 사용하는 경우에는 엔티티명에 파일명을 기록함

제 목 : 데이터 요소(테이블) 명세서	단 계 : 업무 분석 및 설계	작성자 :
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 2/6	승 인 : (인)

번호	엔티티명 (또는 파일명)	한글 : 관리자 영문 : Admin		설 명	특 성		
	한 글 명	영 문 명	필 드 명		Type	Length	Key
1	관리자 번호	ADMIN NUMBER	A_NO	관리자 고유번호	int	—	PK
2	아이디	ID	A_ID	관리자 아이디	char	20	
3	패스워드	PASSWORD	A_PW	관리자 패스워드	char	20	
4	이름	NAME	A_NAME	관리자 이름	char	20	
5	이메일	E MAIL	A_EMAIL	관리자 이메일	char	50	
6	관리자 가입일자	ADMIN JOIN DATE	A_JOIN_DATE	관리자 가입일자	char	8	

Version 1.0

※ 작성 방법 : 파일을 사용하는 경우에는 엔티티명에 파일명을 기록함

제 목 : 데이터 요소(테이블) 명세서	단 계 : 업무 분석 및 설계	작성자 :
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 3/6	승 인 : (인)

번호	엔티티명 (또는 파일명)	한글 : 그룹 영문 : Group		설 명	특 성		
	한 글 명	영 문 명	필 드 명		Type	Length	Key
1	그룹 번호	GROUP NUMBER	G_NUM	그룹 고유번호	int	—	PK
2	그룹 이름	GROUP NAME	G_NAME	그룹 이름	char	20	
3	그룹원 수	MEMBER COUNT	G_MEM_COUNT	총 그룹원 수	int	—	
4	그룹 생성 날짜	GROUP CREATE DATE	G_CREATE_ DATE	그룹 생성 날짜	char	8	
5	팀장 번호	LEADER NUMBER	G_LED_NUM	팀장의 사용자 고유번호	int	—	FK

Version 1.0

※ 작성 방법 : 파일을 사용하는 경우에는 엔티티명에 파일명을 기록함

제 목 : 데이터 요소(테이블) 명세서	단 계 : 업무 분석 및 설계	작성자 :
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 4/6	승 인 : (인)

번호	엔티티명 (또는 파일명)	한글 : 파일 영문 : File		설 명	특 성		
	한 글 명	영 문 명	필 드 명		Type	Length	Key
1	파일 번호	FILE NUMBER	F_NUM	파일 고유번호	int	—	PK
2	파일 이름	FILE NAME	F_NAME	파일 이름	char	50	
3	파일 형식	FILE FORMAT	F_FORM	파일 형식	char	10	
4	파일 크기	FILE SIZE	F_SIZE	파일 크기	int	—	
5	업로드 날짜	UPLOAD DATE	F_UP_DATE	첫 업로드 일자	char	8	FK
6	최종수정 날짜	LAST UPDATE DATE	F_LAST_DATE	최종수정 일자	char	8	
7	업로더	UPLOADER	F_UP_USER	업로드 한 사용자 번호	int	—	
8	업로드 그룹	UPLODE GROUP	F_UP_GROUP	업로드 한 그룹 번호	int	—	
9	파일 위치	FILE LOCATION	F_LOCATION	서버 내 파일 위치	char	100	
10	파일 중요도	FILE IMPORTANCE	F_IMPORT	파일 중요도	int	—	

Version 1.0

※ 작성 방법 : 파일을 사용하는 경우에는 엔티티명에 파일명을 기록함

제 목 : 데이터 요소(테이블) 명세서	단 계 : 업무 분석 및 설계	작성자 :
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 5/6	승 인 : (인)

번호	엔티티명 (또는 파일명)	한글 : 그룹원 영문 : Group member		설 명	특 성		
	한 글 명	영 문 명	필 드 명		Type	Length	Key
	그룹 번호	GROUP NUMBER	G_NUM	속한 그룹 번호	int	—	PK,FK
	사용자 번호	USER NUMBER	U_NUM	속한 사용자 번호	int	—	PK,FK
	그룹 이름	GROUP NAME	G_NAME	속한 그룹의 이름	char	20	FK
	그룹 가입 날짜	GROUP JOIN DATE	G_J_DATE	그룹 가입일자	char	8	
	팀장 여부	LEADER CHECK	G_LEAD_ CHECK	팀장 여부 확인	boolean	—	

Version 1.0

※ 작성 방법 : 파일을 사용하는 경우에는 엔티티명에 파일명을 기록함

제 목 : 데이터 요소(테이블) 명세서	단 계 : 업무 분석 및 설계	작성자 :
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 6/6	승 인 : (인)

번호	엔티티명 (또는 파일명)	한글 : 그룹 파일 권한 영문 : Group Permission		설 명	특 성		
	한 글 명	영 문 명	필 드 명		Type	Length	Key
	그룹 번호	GROUP NUMBER	G_NUM	그룹 고유번호	int	—	PK,FK
	파일 번호	FILE NUMBER	F_NUM	파일 고유번호	int	—	PK,FK
	읽기 권한	READ PERMISSION	P_READ	그룹의 읽기 권한	boolean	—	
	다운로드 권한	DOWNLOAD PERMISSION	P_DOWN	그룹의 다운로드 권한	boolean	—	
	수정 권한	UPDATE PERMISSION	P_UPDATE	그룹의 수정 권한	boolean	—	
	삭제 권한	DELETE PERMISSION	P_DELETE	그룹의 삭제 권한	boolean	—	

Version 1.0

※ 작성 방법 : 파일을 사용하는 경우에는 엔티티명에 파일명을 기록함

제 목 : 기능/엔터티 관계 CRUD	단 계 : 업무 분석 및 설계	작성자 : B조
시스템 :	업 무 : 논리적 데이터 분석 및 설계	작성일 : 2022.11.13.
버 전 :	페이지 : 1/1	승 인 : (인)

기능명 \ 엔터티명	사용자	관리자	그룹	파일	그룹원	그룹 파일 권한
파일 관리	R		R	A	R	A
사용자 설정(관리자)	A	A	R		A	
그룹 관리(관리자)	R		A		A	A
권한 관리(관리자)	R		R	A	R	A

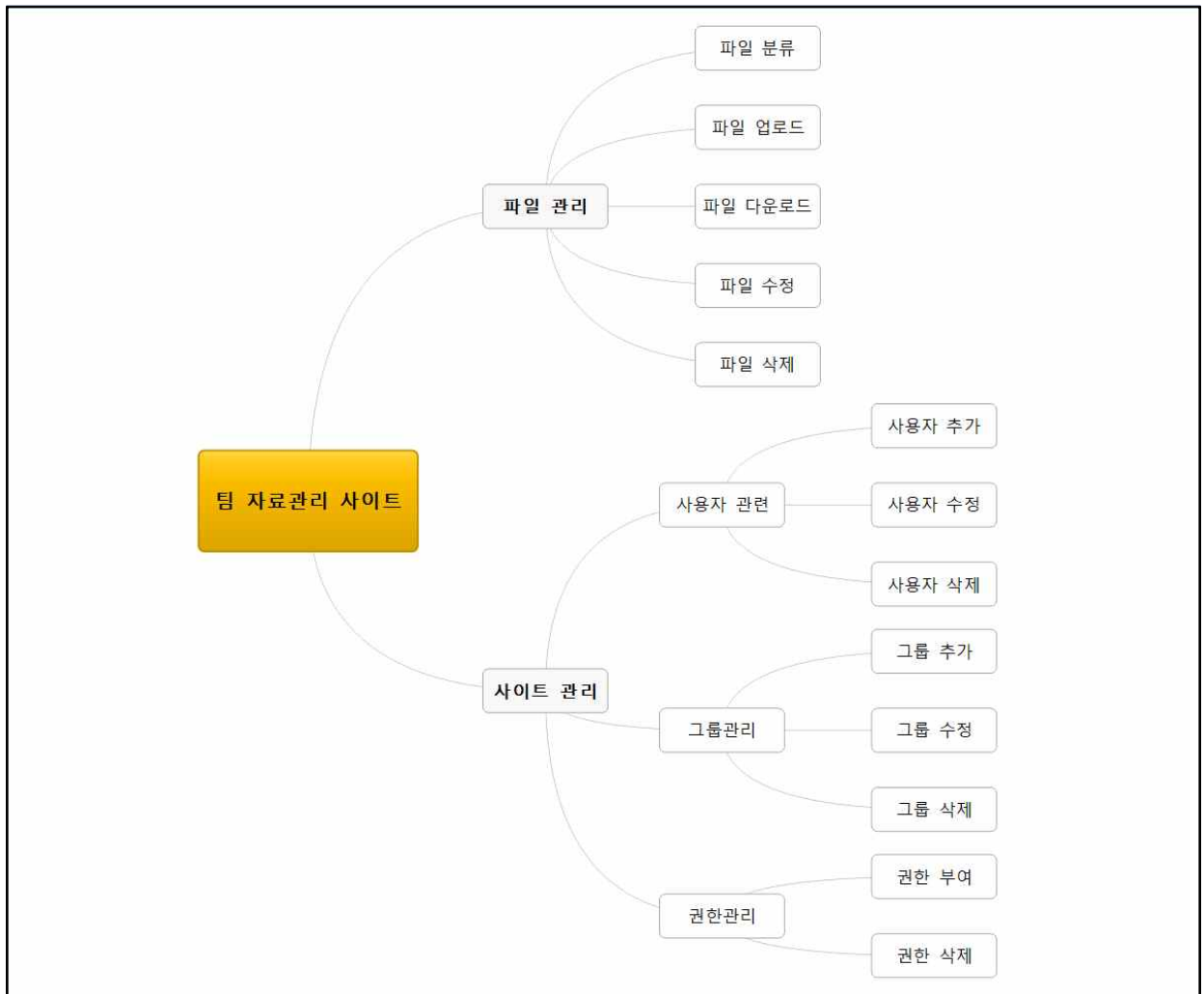
기능분해도

과목명	JSP와 서블릿
연구주제명	팀 자료관리 사이트
팀 명	B팀 (MBTI조)
팀 원	장민관, 권진성, 박주형, 최정규

2022. 10. 14.

제 목 : 메뉴 구조도	단 계 : 업무 분석 및 설계	작성자 : B팀
시 스템 : 팀 자료관리 사이트	업 무 : 기능 설계	작성일 : 2022.10.14.
버 전 : 1.0	페이지 : 1/1	승 인 : (인)

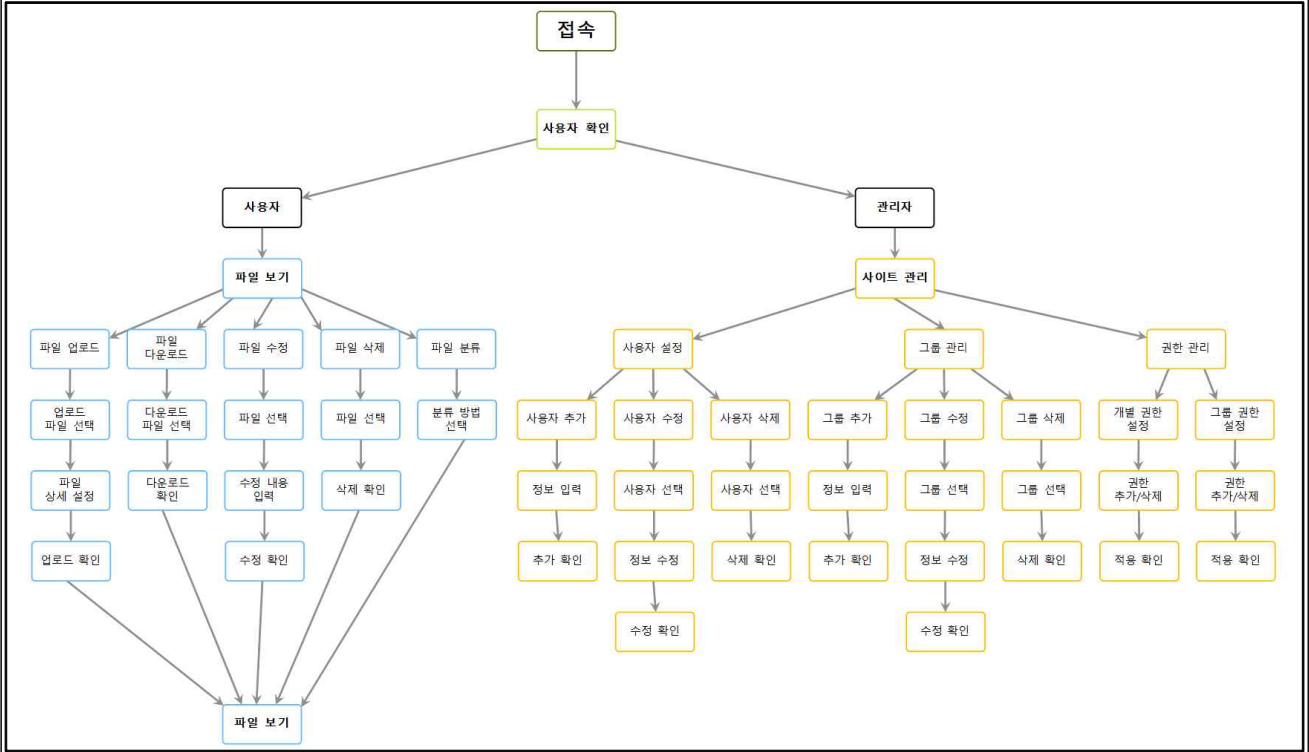
□ 메뉴구조도



↑ 전체 메뉴구조도

제 목 : 기능 흐름도	단 계 : 업무 분석 및 설계	작성자 : B팀
시 스템 : 팀 자료관리 사이트	업 무 : 기능 설계	작성일 : 2022.10.14.
버 전 : 1.0	페이지 : 1/1	승 인 : (인)

□ 기능 흐름도



↑ 사용자/관리자에 따른 기능 흐름도

2022학년도 2학기 JSP와 서블릿

사용자 화면 설계서

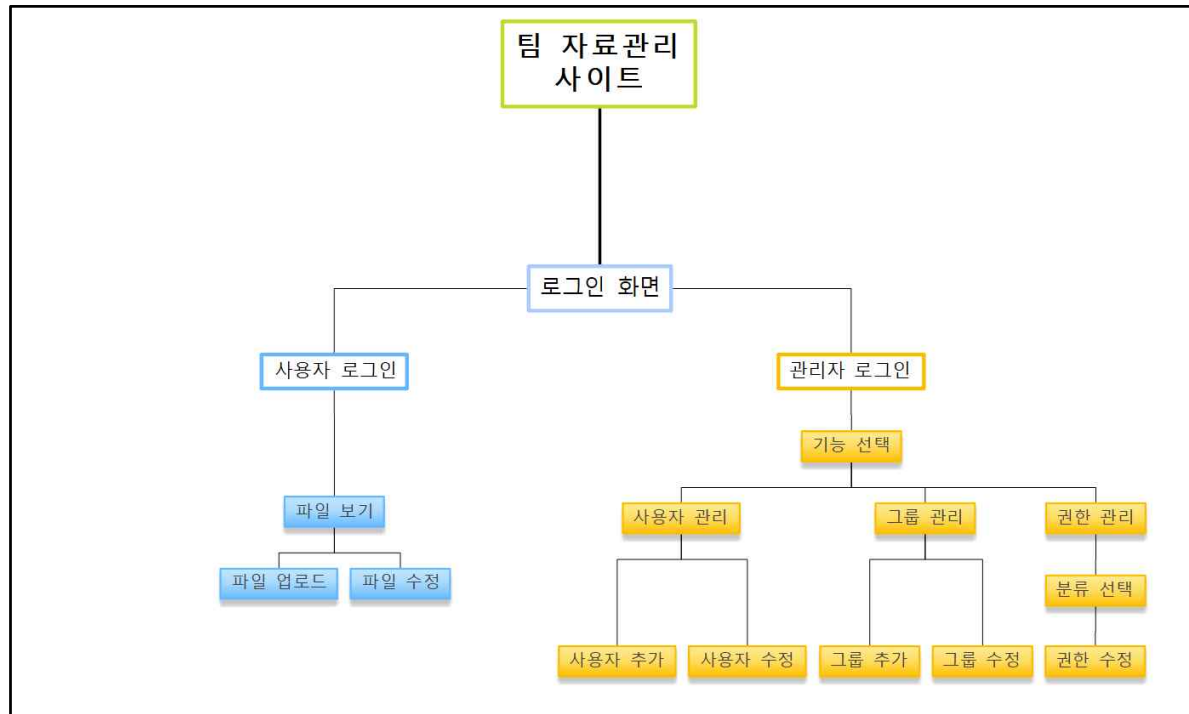
- 팀 자료관리 사이트 -

2022년 10월 18일

팀 명 : B팀 (MBTI 조)

작 성 자 : 장민관, 권진성, 박주형, 최정규

Menu Structure



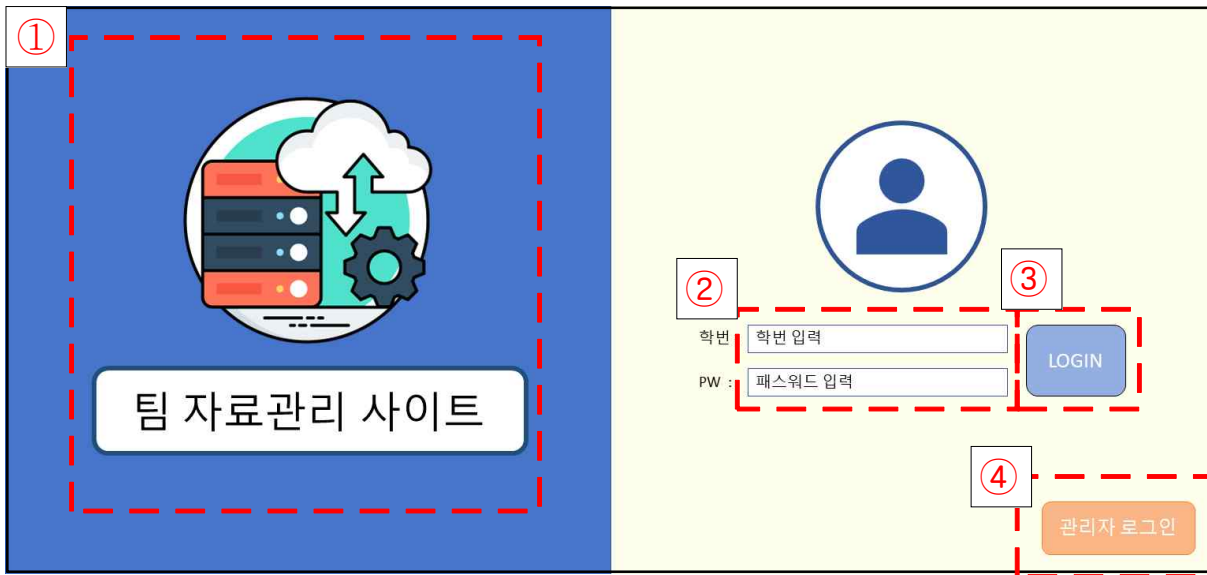
List of Screen

주메뉴	부메뉴	Screen ID	Page Title	Description
로그인	사용자 로그인	l-001	사용자 로그인	사용자의 로그인 화면
	관리자 로그인	l-002	관리자 로그인	관리자의 로그인 화면
파일 관리	파일 리스트	f-001	파일 보기	파일 보기 화면
	파일 정보	f-002	파일 정보	파일 정보 화면
	파일 업로드	f-003	파일 업로드	파일 업로드 화면
	파일 수정	f-004	파일 수정	파일 수정 화면
기능 선택	기능 선택	s-001	관리자 페이지	관리자 기능 선택 화면
사용자 관리	사용자 리스트	u-001	사용자 관리	사용자 리스트 화면
	사용자 추가	u-002	사용자 추가	사용자 추가 화면
	사용자 정보	u-003	사용자 정보	사용자 정보 화면
	사용자 수정	u-004	사용자 수정	사용자 수정 화면
그룹 관리	그룹 리스트	g-001	그룹 관리	그룹 리스트 화면
	그룹 정보	g-002	그룹 정보	그룹 정보 화면
	그룹 추가	g-003	그룹 추가	그룹 추가 화면
	그룹 수정	g-004	그룹 수정	그룹 수정 화면
권한 관리	권한 분류 선택	a-001	분류 선택	권한 분류를 선택하는 화면
	사용자 권한 설정	a-002	사용자 권한설정	관리자의 사용자 권한설정 화면
	그룹 권한 설정	a-003	그룹 권한 설정	관리자의 그룹 권한 설정 화면

사용자화면정의서(UI설계)

ID	I-001	화면 명	사용자 로그인
화면 설명	사용자를 확인하고 접속하는 화면 (사이트의 첫 페이지)		
관련요구사항			

[화면 구성]

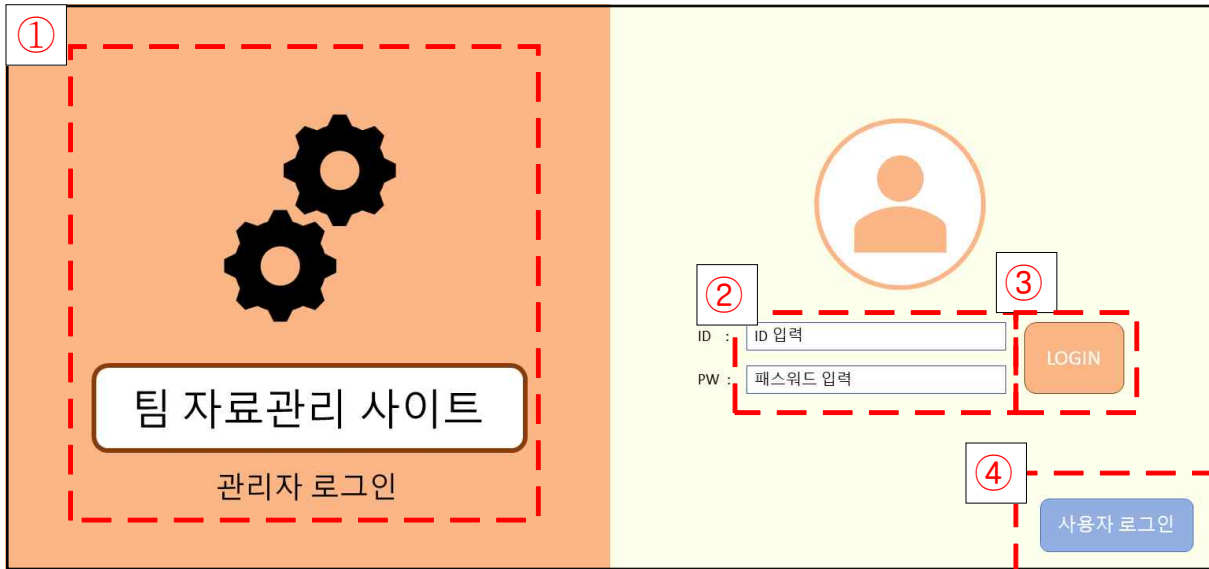


[화면 상세 설명]

- ① : 사이트의 로고 및 타이틀
- ② : 사용자의 학번과 패스워드를 입력하는 란
- ③ : 로그인 버튼
- ④ : 관리자 접속을 위한 로그인 페이지로 이동 버튼

ID	I-002	화면 명	관리자 로그인
화면 설명	관리자를 확인하고 접속하는 화면		
관련요구사항			

[화면 구성]



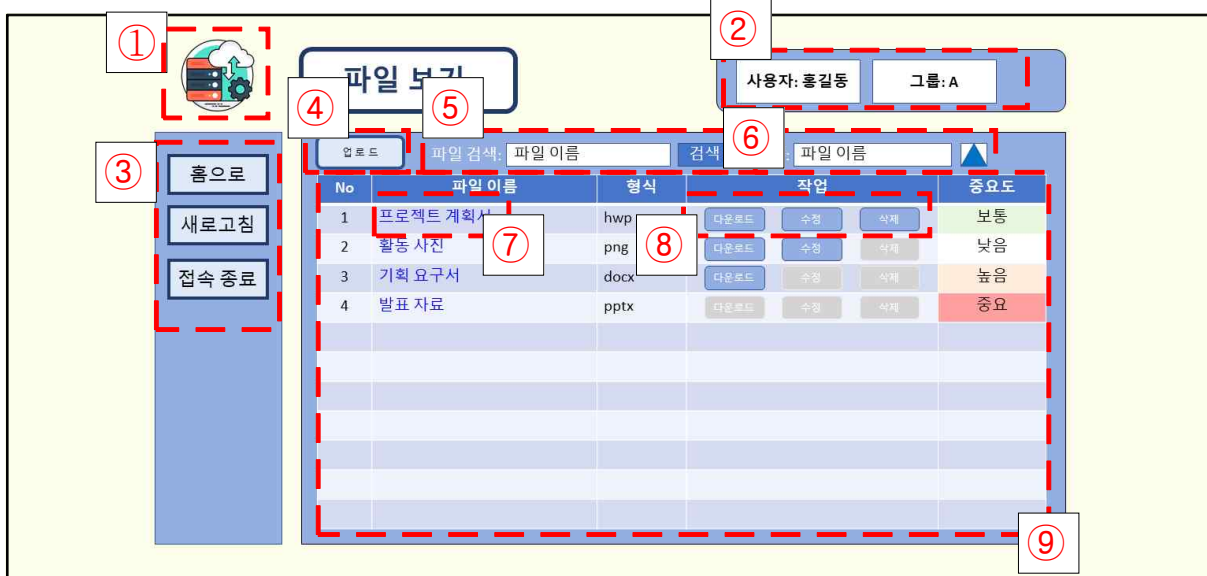
[화면 상세 설명]

- ① : 관리자 로그인 페이지 로고 및 사이트 타이틀, 현재 페이지 안내
- ② : 관리자의 ID와 패스워드를 입력하는 란
- ③ : 관리자 로그인 버튼
- ④ : 사용자 접속을 위한 로그인 페이지로 이동 버튼

사용자 페이지와 구분 짓기 위해 테마색에 차이를 둔다.

ID	f-001	화면 명	파일 보기
화면 설명	사용자의 파일 보기 화면		
관련요구사항			

[화면 구성]



[화면 상세 설명]

- ① : 사이트 로고
- ② : 접속된 사용자의 이름 및 그룹 표시창
- ③ : 작업 메뉴
- ④ : 파일 업로드 버튼(파일 업로드 페이지로 이동)
- ⑤ : 파일 검색 (검색 버튼을 누를시 아래 리스트에 검색 내용이 표시됨)
- ⑥ : 파일 분류 (사용자의 선택에 따라 아래 리스트의 내용이 변경됨)
- ⑦ : 파일 정보 보기 (클릭 시 해당파일의 정보 보기 페이지로 이동)
- ⑧ : 해당 파일의 작업 버튼
- ⑨ : 파일 리스트 표시 창

ID	f-002	화면 명	파일 정보
화면 설명	선택 된 파일의 정보 보기 화면		
관련요구사항			

[화면 구성]

파일 정보

사용자: 홍길동 그룹: A

①

“프로젝트 계획서” 파일 정보

파일 이름	프로젝트 계획서
파일 형식	hwp
업로드 날짜	2022.10.18
수정 날짜	2022.10.18
권한	다운/수정/삭제
업로드 한 사람	팀원2
업로드 그룹	A

②

다운로드 수정 삭제

[화면 상세 설명]

- ① : 선택된 파일의 정보를 표시
- ② : 사용자의 학번과 패스워드를 입력하는 란

ID	f-003	화면 명	파일 업로드 화면
화면 설명	파일을 업로드 하기 위한 입력 페이지		
관련요구사항			

[화면 구성]

파일 업로드

사용자: 홍길동 그룹: A

① 클릭하여 파일 첨부

파일 번호:

파일 이름:

파일 형식:

파일 권한:

사용자	다운로드	다운로드/수정	다운로드/수정/삭제
팀원 1		✓	
팀원 2	✓		
팀원 3			✓

파일 설명:

③ 업로드

[화면 상세 설명]

- ① : 업로드 파일 선택 버튼 (클릭 시 파일 첨부 창 발생)
- ② : 업로드 할 파일의 정보를 설정 (회색칸은 수정 불가)
- ③ : 업로드 버튼 (서버에 파일을 업로드한다)

ID	f-004	화면 명	파일 수정
화면 설명	선택된 파일을 수정하기 위한 페이지		
관련요구사항			

[화면 구성]

파일 수정

사용자: 홍길동 그룹: A

① 클릭하여 파일 첨부

②

파일 번호 : 파일 번호 (수정 불가)

파일 이름 : 파일 이름

파일 형식 : 파일 형식 (수정 불가)

파일 권한 :

사용자	다운로드	다운로드/수정	다운로드/수정/삭제
팀원 1		✓	
팀원 2	✓		
팀원 3			✓

파일 설명 : 파일 설명 작성란

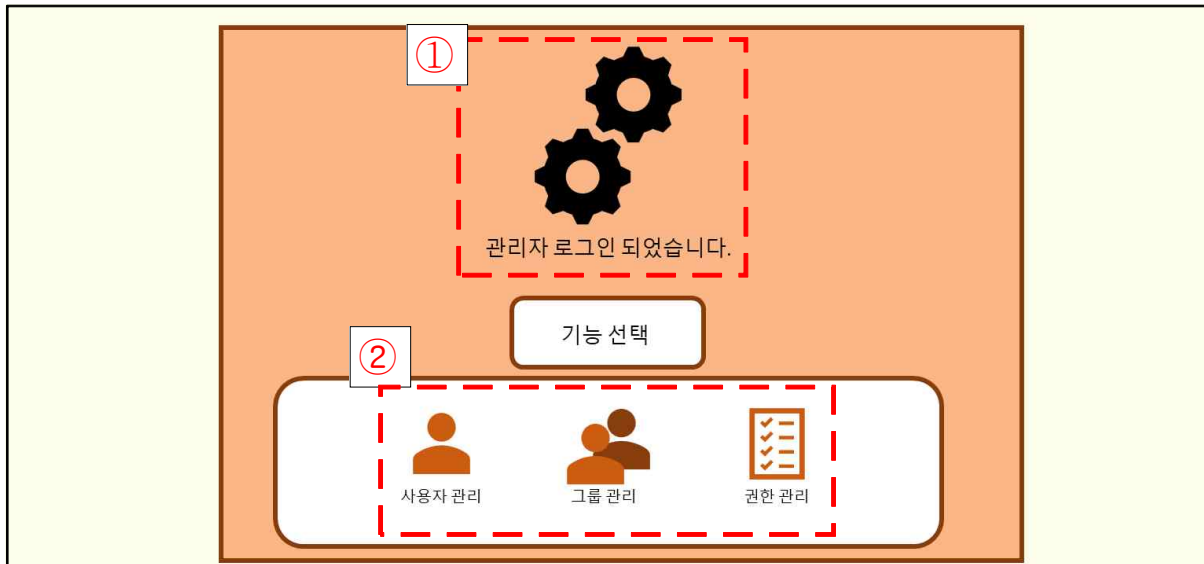
③ 수정

[화면 상세 설명]

- ① : 수정 파일 첨부 버튼
- ② : 수정할 파일의 내용 입력
- ③ : 수정 버튼

ID	s-001	화면 명	관리자 페이지
화면 설명	관리자 접속 후 관리자의 기능을 선택하기 위한 페이지		
관련요구사항			

[화면 구성]



[화면 상세 설명]

- ① : 관리자 로그인 확인 문구
- ② : 관리자 기능 선택 메뉴 (선택 시 각 기능의 페이지로 이동)

ID	u-001	화면 명	사용자 관리
화면 설명	사이트 사용자의 리스트 출력 페이지		
관련요구사항			

[화면 구성]



[화면 상세 설명]

- ① : 관리자 로그인 확인 문구
- ② : 관리자 기능 선택 메뉴 (선택 시 각 기능의 페이지로 이동)

ID	u-002	화면 명	사용자 추가
화면 설명	사이트의 사용자를 추가하는 페이지		
관련요구사항			

[화면 구성]

[화면 상세 설명]

- ① : 추가할 사용자의 정보 입력란
- ② : 팀장여부 확인 체크박스 (체크 시 해당 그룹의 팀장이 된다)
- ③ : 해당 사용자의 권한을 체크하는 버튼
- ④ : 사용자 추가 버튼

ID	u-003	화면 명	사용자 정보
화면 설명	선택된 사용자의 정보 출력 페이지		
관련요구사항			

[화면 구성]

①

홍길동 사용자 정보

사용자 이름	홍길동
학 번	20202222
추가 날짜	2022.10.18
수정 날짜	2022.10.18
권한	다운/수정/삭제
그룹	A
그룹 팀장 여부	O

②

수정 삭제

[화면 상세 설명]

- ① : 선택된 사용자의 정보를 보여주는 창
- ② : 선택된 사용자의 '수정', '삭제'

ID	u-004	화면 명	사용자 수정
화면 설명	선택된 사용자의 정보를 수정하는 페이지		
관련요구사항			

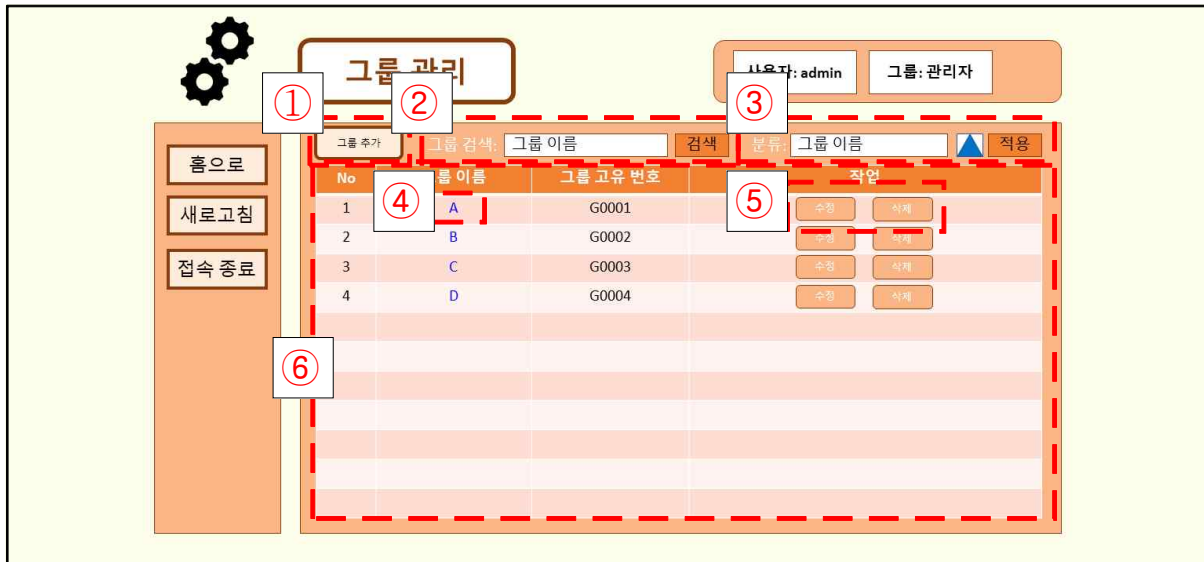
[화면 구성]

[화면 상세 설명]

- ① : 선택 된 사용자의 정보 수정
- ② : 사용자 정보 수정 버튼

ID	g-001	화면 명	그룹 관리
화면 설명	사이트에 존재하는 그룹의 리스트 출력 페이지		
관련요구사항			

[화면 구성]



[화면 상세 설명]

- ① : 새로운 그룹 추가 버튼
- ② : 그룹 검색 기능
- ③ : 그룹 분류 기능 (적용 버튼 클릭 시 해당사항 리스트에 반영)
- ④ : 그룹 이름을 클릭 할 시 해당 그룹의 정보 보기 페이지로 이동
- ⑤ : 해당 그룹 수정 또는 삭제 버튼
- ⑥ : 그룹 출력 리스트

ID	g-002	화면 명	그룹 정보
화면 설명	선택된 그룹의 정보를 보여주는 페이지		
관련요구사항			

[화면 구성]



[화면 상세 설명]

- ① : 선택 된 그룹의 정보 표시
- ② : 선택 된 그룹의 수정 또는 삭제 버튼

ID	g-003	화면 명	그룹 추가
화면 설명	그룹을 추가하는 페이지		
관련요구사항			

[화면 구성]

[화면 상세 설명]

- ① : 추가 할 그룹의 정보 입력
- ② : 팀장 체크박스 체크 시 해당 그룹의 팀장 설정
- ③ : 그룹 추가 버튼

ID	g-004	화면 명	그룹 수정
화면 설명	선택된 그룹을 수정하는 페이지		
관련요구사항			

[화면 구성]

[화면 상세 설명]

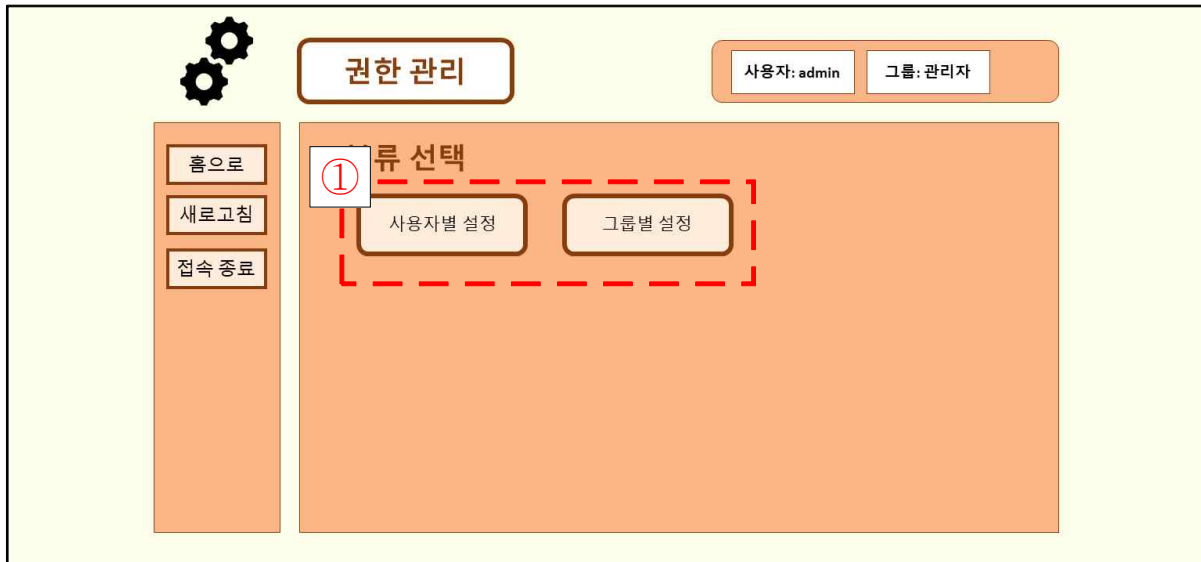
①②③④⑤⑥⑦⑧⑨⑩

① : 선택 된 그룹의 정보 수정란

② : 정보 수정 버튼

ID	a-001	화면 명	사용자 권한 설정
화면 설명	사용자 권한을 설정하는 페이지 (개별적으로 선택)		
관련요구사항			

[화면 구성]



[화면 상세 설명]

① : 사용자별/그룹별 분류 선택 버튼 해당 버튼을 클릭 시 각각의 권한 관리 페이지로 이동

ID	a-002	화면 명	사용자 권한 설정
화면 설명	사용자 권한을 설정하는 페이지 (개별적으로 선택)		
관련요구사항			

[화면 구성]

사용자 권한 설정

사용자: admin 그룹: 관리자

① 사용자 검색: 사용자 이름 ② 사용자 이름 ▲ 적용

③

No	사용자 이름	그룹	학번	권한
1	홍길동	A	20210001	일반
2	이길동	B	20221003	일반
3	이길동	C	20180305	일반
4	삼길동	D	20201015	일반

다운로드 수정 삭제 다운로드 수정 삭제 다운로드 수정 삭제 다운로드 수정 삭제

④

[화면 상세 설명]

- ① : 사용자 검색 기능
- ② : 사용자 분류 기능
- ③ : 사용자 리스트 출력 창
- ④ : 해당 사용자의 권한 설정

ID	a-003	화면 명	그룹 권한 설정 페이지
화면 설명	그룹의 권한을 설정하는 페이지		
관련요구사항			

[화면 구성]



[화면 상세 설명]

- ① : 그룹 검색 기능
- ② : 그룹 분류 기능
- ③ : 그룹 리스트 출력
- ④ : 해당 그룹의 권한 변경

팀 자료 관리 서비스

TFMS : Team File Manage Service

B조 (MBTI조)

JSP와 서블릿 - 김병철 교수님 (목 7,8,9/사)

목차

1. 팀원 소개
2. 주제 소개
3. 목표 및 달성도
4. 세부 실행 내용
5. 구현 결과(+시연)
6. 성과 및 기대효과
7. 느낀점 및 개선사항

팀원소개

팀장. 장민관



프론트엔드, JSP보조,
서버 구현

팀원. 박주형



HTML, CSS, BootsStrap
프론트엔드

팀원. 최정규



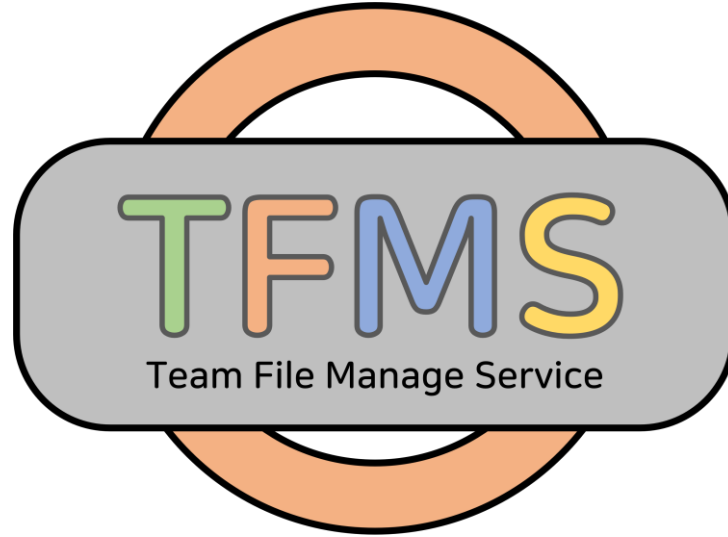
파일관리 백엔드(JSP)

팀원. 권진성



로그인,회원가입,
유저관리 백엔드(JSP)

주제소개

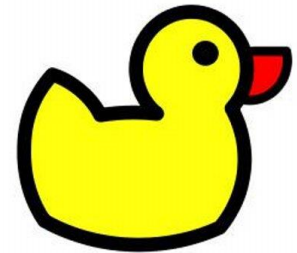
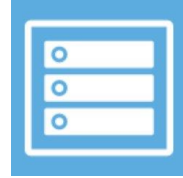
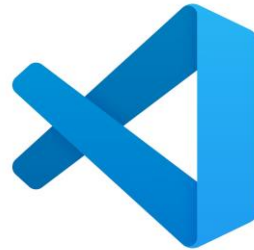
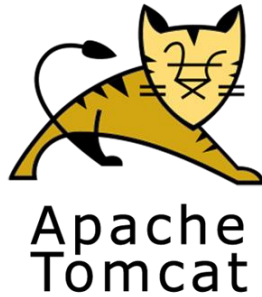
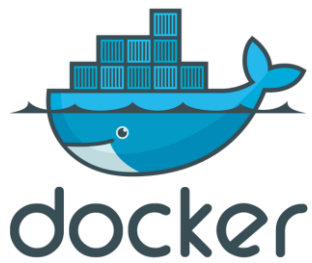


Big : 팀별 파일 관리 서비스

small : 서버를 통해 원거리에서 팀의 파일을 관리하는 서비스
여러 그룹을 참여 및 생성하여 팀별로 파일을 관리한다.
DB와 파일 입출력을 적절히 적용하여 구현한다.

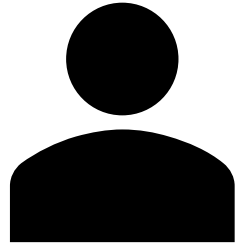
세부 실행 내용

서버 환경 - 사용 프로그램

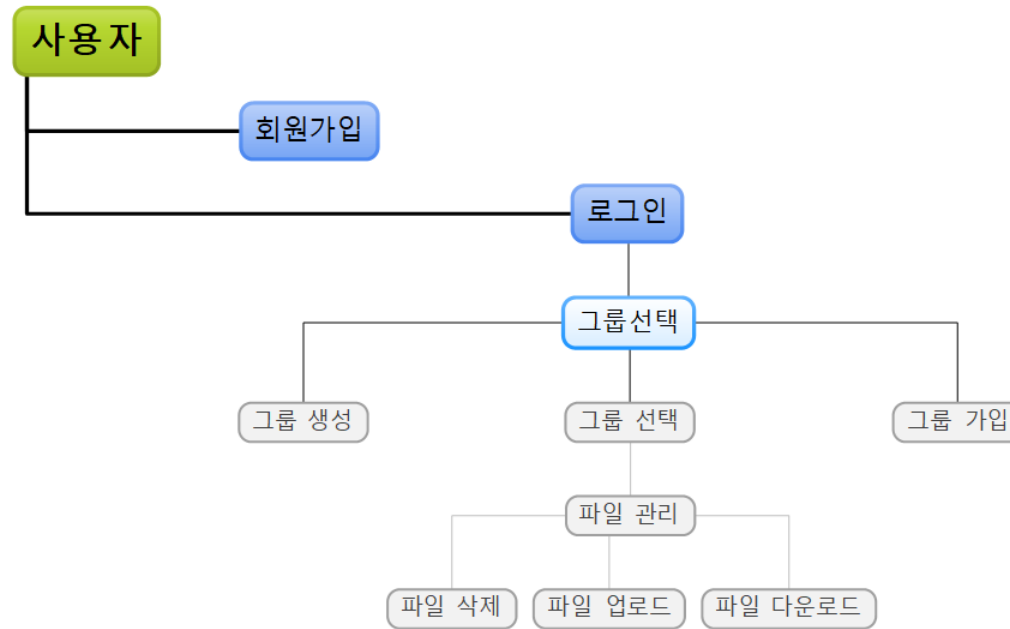


세부 실행 내용

기능구조도 : 사용자 시점

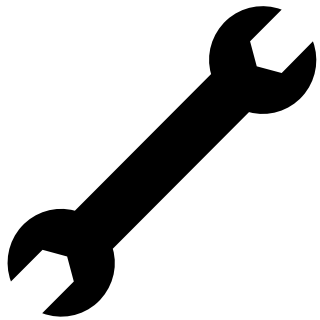


사용자

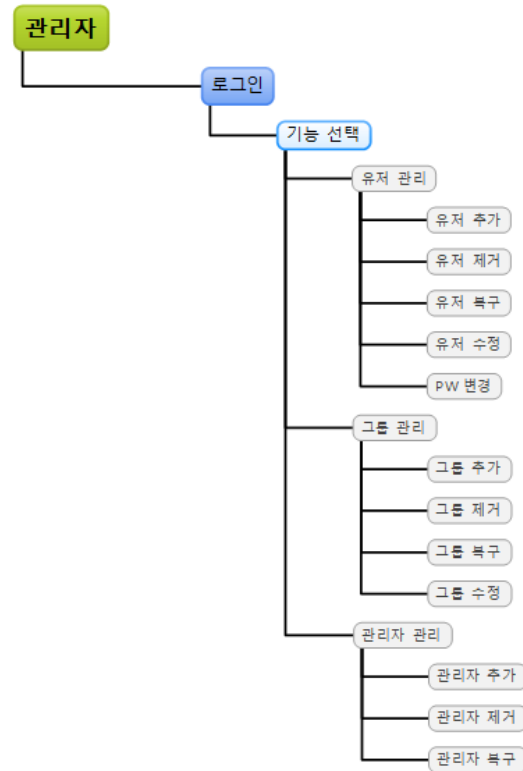


세부 실행 내용

기능구조도 : 관리자 시점

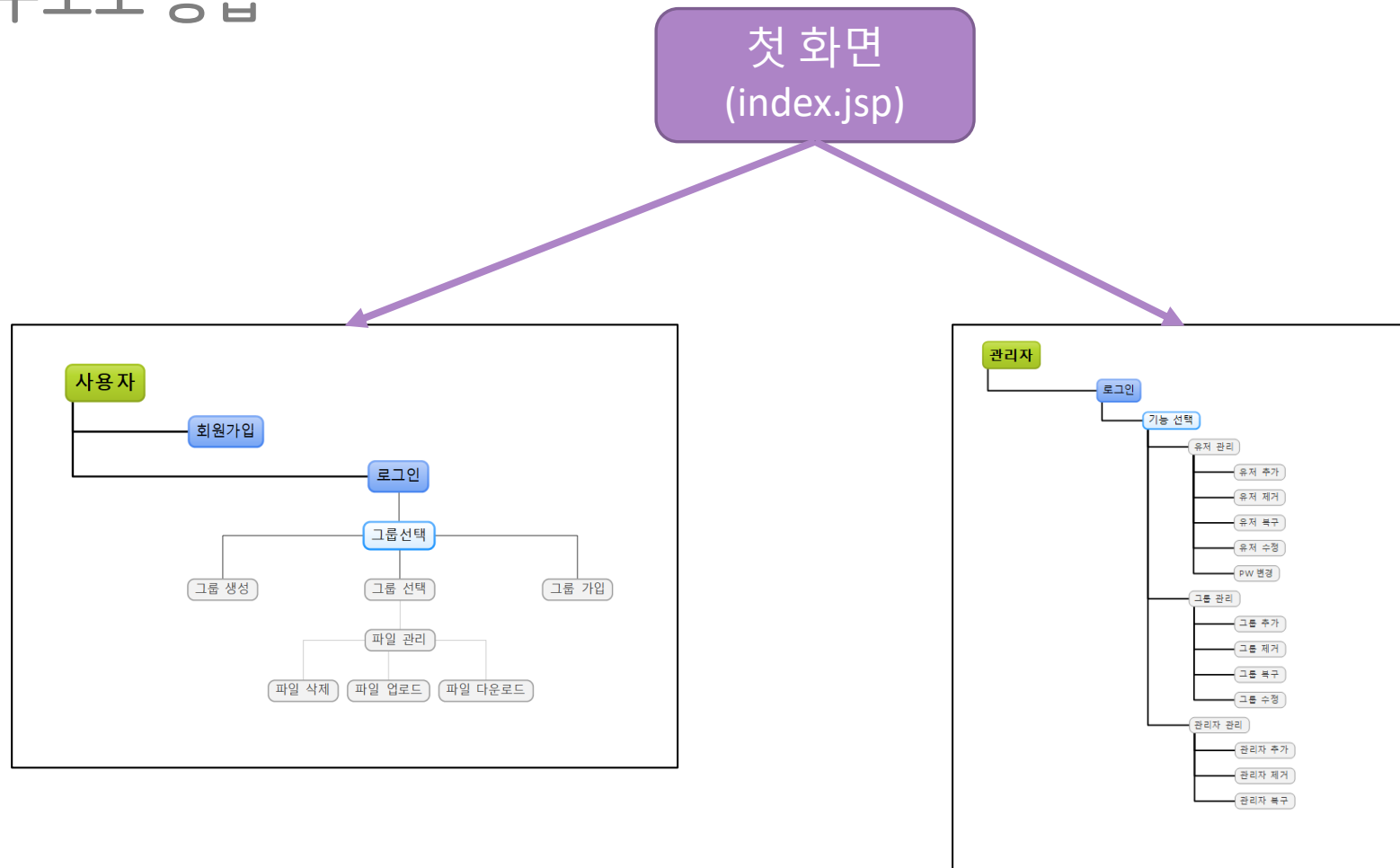


관리자



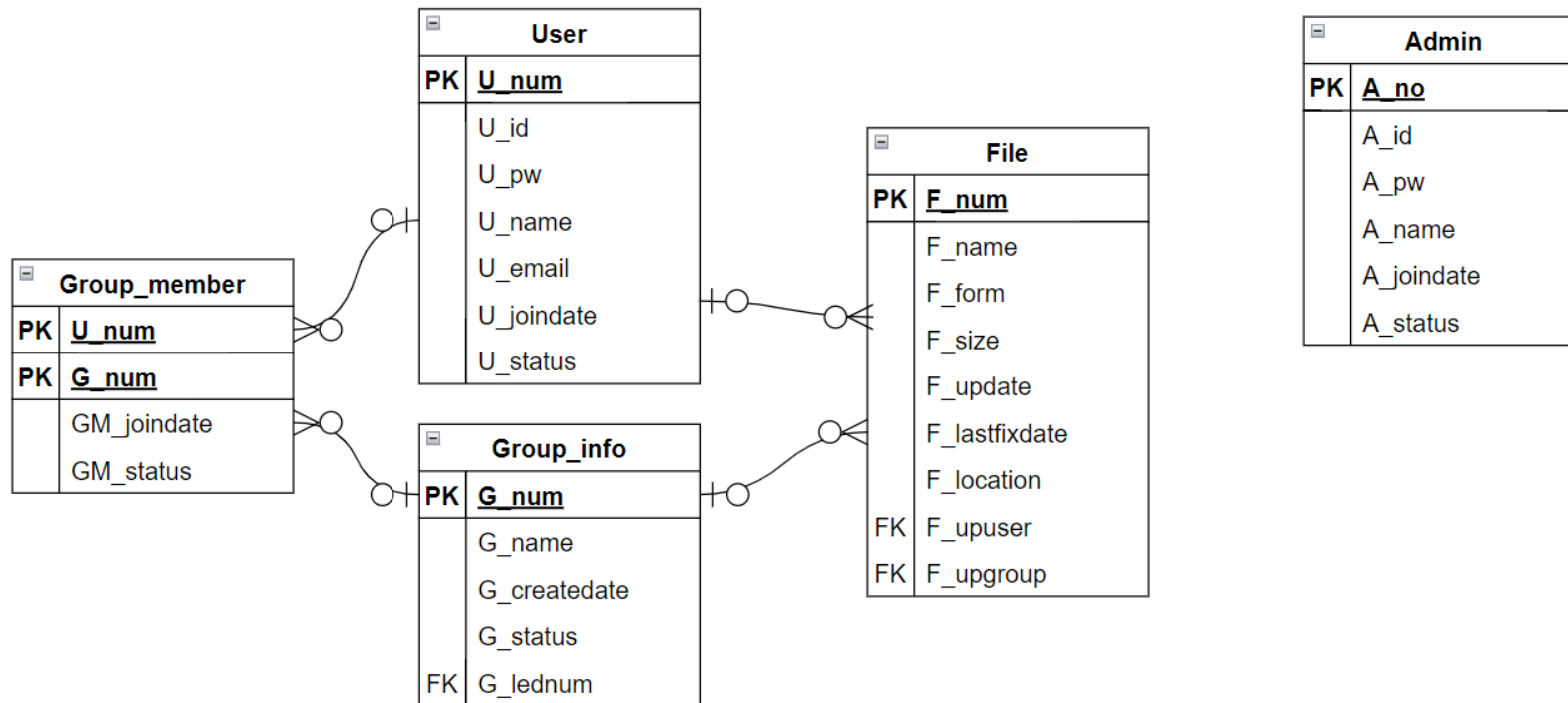
세부 실행 내용

기능구조도 종합



세부 실행 내용

DB 설계



구현 결과

DB구성

테이블

<input type="checkbox"/>	테이블	엔진?
<input type="checkbox"/>	admin	InnoDB
<input type="checkbox"/>	file	InnoDB
<input type="checkbox"/>	group_info	InnoDB
<input type="checkbox"/>	group_member	InnoDB
<input type="checkbox"/>	user	InnoDB
	총 5개	InnoDB

admin

열	형	주석
A_no	int(11) 자동 증가	
A_id	char(20)	
A_pw	char(20)	
A_name	char(20) NULL	
A_joindate	date NULL	
A_status	tinyint(1) NULL	

file

열	형	주석
F_num	int(11) 자동 증가	
F_name	char(100) NULL	
F_form	char(10) NULL	
F_size	char(10) NULL	
F_update	date NULL	
F_last_fix_date	date NULL	
F_upuser	int(11)	
F_upgroup	int(11)	
F_location	char(100) NULL	

group_info

열	형	주석
G_num	int(11) 자동 증가	
G_name	char(20) NULL	
G_createdate	date NULL	
G_lednum	int(11) NULL	
G_status	tinyint(1) NULL	

group_member

열	형	주석
G_num	int(11)	
U_num	int(11)	
GM_joindate	date NULL	
GM_status	tinyint(4) NULL [1]	

user

열	형	주석
U_num	int(11) 자동 증가	
U_id	char(30)	
U_pw	char(30)	
U_name	char(20) NULL	
U_email	char(50) NULL	
U_joindate	date NULL	
U_status	tinyint(1) NULL	



백석대학교
BAEKSEOK UNIVERSITY

구현 결과

기본 디자인



TFMS
Team File Manage System

로그인

회원가입

관리자 로그인

Team B (MBTI 조)



TFMS:관리자

유저 관리

관리자명 : 관리자01 [로그아웃](#)

[이전으로](#) [유저 추가](#) [유저 복구](#)

번호	유저 고유번호	유저 아이디	이름	이메일	가입일자			
1	27	t1	test11	-	2022-12-18	수정	PW 변경	제거
2	28	aaa	asd	-	2022-12-18	수정	PW 변경	제거
3	30	qqqqqq	-	-	2022-12-18	수정	PW 변경	제거
4	31	sdf	-	-	2022-12-18	수정	PW 변경	제거
5	34	cjg721	최정규	cjg721@naver.com	2022-12-20	수정	PW 변경	제거

Team B (MBTI 조)



백석대학교
BAEKSEOK UNIVERSITY

구현 결과

기능 설명 - 회원가입



회원가입

ID

중복확인

PW

PW확인

사용자 이름

이메일

가입하기

[이전으로](#)

ID 확인 - Chrome

주의 요함 | tfms.duckdns.org:81/do/UIDCheckDo.jsp?idcheck=t1

ID 중복확인
"t1" 아이디는
중복 아이디로 사용 불가능합니다.
[이전으로](#)

ID 확인 - Chrome

주의 요함 | tfms.duckdns.org:81/do/UIDCheckDo.jsp?idcheck=t2

ID 중복확인
"t2" 아이디는
사용가능합니다.
[사용하기](#) [이전으로](#)

tfms.duckdns.org:81 내용:

가입하시겠습니까?

확인

취소

tfms.duckdns.org:81 내용:

패스워드 확인이 틀립니다.

확인

tfms.duckdns.org:81 내용:

가입되었습니다.

확인

구현 결과

기능 설명 - 회원가입

```
LocalDate now = LocalDate.now();
String uid = request.getParameter("user_id");
String upw = request.getParameter("user_pw");
String uname = request.getParameter("user_name");
String uemail = request.getParameter("user_email");
String udate = now.toString();//formatter.format(now);
String ustat = "1";
if(uname == ""){uname = null;}
if(uemail == ""){uemail = null;}
StringBuffer sql = new StringBuffer("insert into user(U_id,U_pw,U_name,U_email,U_joindate,U_status)");
sql.append("values(?,?,?,?,?,?)");
PreparedStatement pstmt = conn.prepareStatement(sql.toString());
pstmt.setString(1, uid);
pstmt.setString(2, upw);
pstmt.setString(3, uname);
pstmt.setString(4, uemail);
pstmt.setString(5, udate);
pstmt.setString(6, ustat);
pstmt.executeUpdate();
```

U_num	U_id	U_pw	U_name	U_email	U_joindate	U_status
27	t1	t1	test11	NULL	2022-12-18	1
28	aaa	aa	asd	NULL	2022-12-18	1
29	zzz	ddd	NULL	NULL	2022-12-18	0
30	qqqqqq	qqq	NULL	NULL	2022-12-18	1
31	sdf	sdf	NULL	NULL	2022-12-18	1
33	t2	qqqq	NULL	NULL	2022-12-20	1

회원가입 창에서 입력한 내용들을 가져오고 현재 시간을 가져온 후 유저 상태(U_status)를 1로 저장하게 됩니다. 이후 가져온 내용을 DB에 저장하게 된다.



백석대학교
BAEKSEOK UNIVERSITY

구현 결과

기능 설명 - 로그인



tfms.duckdns.org:81 내용:

로그인 할까요?

확인

취소

tfms.duckdns.org:81 내용:

로그인 실패
(아이디 또는 비밀번호가 다릅니다.)

tfms.duckdns.org:81 내용:

로그인 성공
(test11님 환영합니다.)

구현 결과

기능 설명 - 로그인

```
function confirmCheck(){
    var idvalue = document.getElementsByName("user_id").value;
    var pwvalue = document.getElementsByName("user_pw").value;
    if(idvalue == ""){
        alert("ID를 입력해주세요.");
        return false;
    }
    else{
        if(pwvalue == ""){
            alert("PW를 입력해주세요.");
            return false;
        }
    }
    var confirmresult = confirm("로그인 할까요?");
    if(confirmresult==false){
        return false;
    }else{return true;}
}
```

```
if ("<%=idResult%>" == 0) {
    alert("로그인 실패\n(아이디 또는 비밀번호가 다릅니다.)");
    window.history.back();
}
else {
    if ("<%=statusResult%>" == 0) {
        alert("로그인 실패\n(탈퇴 처리된 회원입니다.)");
        window.history.back();
    }
    else {
        if ("<%=upw%>" == "<%=pwResult%>") {
            localStorage.setItem('unum', "<%=usernumResult%>");
            sessionStorage.setItem('uname', "<%=nameResult%>");
            sessionStorage.setItem('uemail', "<%=emailResult%>");
            sessionStorage.setItem('ujoindate', "<%=userjdateResult%>");
            alert("로그인 성공\n(<%=nameResult%>님 환영합니다.)");
            location.href="/userside/Gselect.jsp?Unum=<%=usernumResult%>";
        }
        else {
            alert("로그인 실패\n(아이디 또는 비밀번호가 다릅니다.)");
            window.history.back();
        }
    }
}
```

아이디와 비밀번호 창이 비어있는지 확인과
로그인 실패에 관련된 코드 및 입력된 정보가 세션에 들어가는 코드



백석대학교
BAEKSEOK UNIVERSITY

구현 결과

기능 설명 - 로그아웃



그룹 선택

유저이름 : test11 [로그아웃](#)

tfms.duckdns.org:81 내용:
로그아웃 하시겠습니까?

확인

취소

로그아웃 처리시 사용자 확인 작업 이후 세션스토리지 내용을 지우고 로그아웃 합니다.

구현 결과

DB데이터 리스트화

		그룹생성	그룹가입			
번호	그룹 번호	그룹 이름	생성 날짜	팀장 번호		
1	24	test1	2022-12-18	-	선택	
2	27	test3	2022-12-18	27	선택	
3	28	1234	2022-12-20	27	선택	
4	29	aaafggdd	2022-12-20	27	선택	
5	30	aaggvvcc	2022-12-20	27	선택	
6	31	qqtdaffa	2022-12-20	27	선택	
7	32	1234567788	2022-12-20	27	선택	
8	34	2234	2022-12-20	27	선택	
9	40	555667	2022-12-20	27	선택	
10	41	11232415	2022-12-20	27	선택	
11	43	333333	2022-12-20	27	선택	
12	44	ccc	2022-12-20	27	선택	

```
<%
// 세션생성
//session = request.getSession();

String unum = request.getParameter("Unum");
String sql1 = "Select count(G_num) as Gcount from group_member";
String sql2 = "Select * from group_info where G_status = 1 and
//sql 즉시 실행
Statement stmt1 = conn.createStatement();
//Select SQL문
ResultSet rs1 = stmt1.executeQuery(sql1);
rs1.next();
int gcount = Integer.parseInt(rs1.getString("gcount"));

Statement stmt2 = conn.createStatement();
//Select SQL문
ResultSet rs2 = stmt2.executeQuery(sql2);
//[0]: gnum [1]: gname [2]: gdate [3]: gleadnum
String[][] GroupList = new String[gcount][4];

//GroupList
for(int i=0;i<gcount;i++){
    rs2.next();
    GroupList[i][0] = rs2.getString("G_num");
    GroupList[i][1] = rs2.getString("G_name");
    GroupList[i][2] = rs2.getString("G_createdate");
    GroupList[i][3] = rs2.getString("G_lednum");
    if(GroupList[i][0] == null){GroupList[i][0] = "-";}
    if(GroupList[i][1] == null){GroupList[i][1] = "-";}
    if(GroupList[i][2] == null){GroupList[i][2] = "-";}
    if(GroupList[i][3] == null){GroupList[i][3] = "-";}
}
```

```
<tr>
    <th>번호</th>
    <th>그룹 번호</th>
    <th>그룹 이름</th>
    <th>생성 날짜</th>
    <th>팀장 번호</th>
</tr>
</thead>
<tbody>
    <%
        for(int j=0;j<gcount;j++){
            int count = j;
    <tr>
        <td><%=j+1%></td>
        <td><%=GroupList[j][0]%></td>
        <td><%=GroupList[j][1]%></td>
        <td><%=GroupList[j][2]%></td>
        <td><%=GroupList[j][3]%></td>
        <td><input type="button" value="선택" onclick="groupSelect(<%=GroupList[j][0]%>)"></td>
    </tr>
    <%
        }
    </tbody>
</table>
```

구현 결과

유저/그룹 추가, 제거, 수정, 복구

이전으로 유저 추가 유저 복구								
번호	유저 고유번호	유저 아이디	이름	이메일	가입일자			
1	27	t1	test11	-	2022-12-18	수정	PW 변경	제거
2	28	aaa	asd	-	2022-12-18	수정	PW 변경	제거
3	30	qqqqqq	-	-	2022-12-18	수정	PW 변경	제거
4	31	sdf	-	-	2022-12-18	수정	PW 변경	제거
5	34	cjg721	최정규	cjg721@naver.com	2022-12-20	수정	PW 변경	제거

구현 결과

유저/그룹 추가, 제거, 수정, 복구

추가

```
LocalDate now = LocalDate.now();
String gname = request.getParameter("group_name");
String gleader = request.getParameter("leader_num");
String gdate = now.toString();
String gstatus = "1";
if(gleader == ""){gleader = null;}

//sql
StringBuffer sql = new StringBuffer("insert into group_info(G_name, G_leader, G_date, G_status)");
sql.append("values(?, ?, ?, ?)");
PreparedStatement pstmt = conn.prepareStatement(sql.toString());
pstmt.setString(1, gname);
pstmt.setString(2, gleader);
pstmt.setString(3, gdate);
pstmt.setString(4, gstatus);
pstmt.executeUpdate();
```

수정

```
String gno = request.getParameter("Gno");
String gname = request.getParameter("group_name");
String gleader = request.getParameter("leader_num");
if(gleader == ""){gleader = null;}

//sql
StringBuffer sql = new StringBuffer("update group_info set G_name = ?, G_leader = ?");
PreparedStatement pstmt = conn.prepareStatement(sql.toString());
pstmt.setString(1, gname);
pstmt.setString(2, gleader);
pstmt.executeUpdate();
```

제거

```
String gno = request.getParameter("Gno");

//sql
StringBuffer sql = new StringBuffer("update group_info set G_status = '0'");
sql.append("where G_num = ?");
PreparedStatement pstmt = conn.prepareStatement(sql.toString());
pstmt.setString(1, gno);
pstmt.executeUpdate();
```

복구

```
//sql
StringBuffer sql = new StringBuffer("update group_info set G_status = '1'");
sql.append("where G_num = ?");
PreparedStatement pstmt = conn.prepareStatement(sql.toString());
pstmt.setString(1, gno);
pstmt.executeUpdate();
```



백석대학교
BAEKSEOK UNIVERSITY

구현 결과_파일 리스트

그룹번호 : 27

파일 업로드 [이전으로](#)

번호	파일번호	파일이름	파일형식	파일크기	올린날짜	
1	14	20221220_082050.png	png	35Kb	2022-12-20	<div>다운로드</div> <div>삭제</div>
2	19	Homework_#13_Solution.pdf	pdf	63Kb	2022-12-20	<div>다운로드</div> <div>삭제</div>
3	20	연구노트_0팀_학번_이름_0주차.hwp	hwp	13Kb	2022-12-20	<div>다운로드</div> <div>삭제</div>

1)먼저 sql을 통해서 세션에 저장되어 있는 그룹넘버에 해당하는 파일들의 목록을 전부 가져온다.

2)해당 정보들을 반복을 통해 테이블에 리스트 형식으로 보여주고 여러 버튼들에 같이 넘겨줄 값들을 같이 넣어준다

3)파일 이름에는 링크를 걸어 파일을 다운로드할 수 있는 jsp파일로 이동한다.

구현 결과_파일 리스트_소스코드

```
sql = "select * from file where F_upgroup = " + gnum;

int rowCount = 1;
try {
    pstmt = conn.prepareStatement(sql);

    ResultSet result = pstmt.executeQuery();
}%
<%@ include file="/topBarUser.jsp" %>
<script>
    window.onload = function(){
        document.getElementById("barsiteinfo").innerText = "TFMS";
        document.getElementById("bartitle").innerText = "파일관리";
        document.getElementById("baruser").innerText = "유저이름 : " + sessionStorage.getItem("uname");
        document.getElementById("bargroup").innerText = "그룹번호 : " + <%=gnum%>;
    }
</script>
<h3>그룹번호 : <%=gnum%></h3>
<input type="button" value="파일 업로드" onclick="location.href='/userside/FUpload.jsp?Unum=<%=unum%>&Gnum=<%=gnum%>'">
<a href="/userside/Gselect.jsp?Unum=<%=unum%>">이전으로</a>
<table style="margin-left: auto; margin-right: auto;" class='table-bordered table-sm'>
    <thead>
        <tr>
            <th>번호</th>
            <th>파일번호</th>
            <th>파일이름</th>
            <th>파일형식</th>
            <th>파일크기</th>
            <th>올린날짜</th>
        </tr>
    </thead>
    <tbody>
```

```
<tbody>
    <% while(result.next()){ %>
    <tr>
        <td><%= rowCount %></td>
        <td><%= result.getInt("F_num") %></td>
        <td><%= result.getString("F_name") %></td>
        <td><%= result.getString("F_form") %></td>
        <td><%= result.getString("F_size") %></td>
        <td><%= result.getString("F_update") %></td>
        <td><input type="button" value="다운로드" onclick="location.href='FDownload_test_Do.jsp?Unum=<%=unum%>&Gnum=<%=gnum%>&Fno=<%= result.getInt("F_num") %>'></td>
        <td><a href="FDelete_test_Do.jsp?Unum=<%=unum%>&Gnum=<%=gnum%>&Fno=<%= result.getInt("F_num") %>">삭제</a></td>
    </tr>
    <%
        rowCount++;
    }
    result.close();
}
catch (Exception e) {
    out.println("[board] 테이블 조회에 문제가 있습니다. <hr>");
    out.println(e.toString());
    e.printStackTrace();
}
finally {
    if (pstmt != null) pstmt.close();
    if (conn != null) conn.close();
}
    %>
</tbody>
</table>
<script src="/scripts.js"></script>
<%@ include file="/bottomBar.jsp" %>
```



구현 결과_파일 업로드

※ 파일 업로드시 한글파일명 사용 불가

파일 선택	20221221_032052.png	업로드
이전으로		

- 1)파일 업로드는 수업시간에 사용했던 cos라이브러리를 이용해서 파일 업로드를 진행하였다.
- 2)먼저 파일정보를 db에 넣을 sql문을 작성하였다.
그다음 그룹넘버를 이름으로 가지는 경로를 넣어서 multipartRequest를 설정해주었다.
- 3) sql에 넣을 요소들을 정리해주었다. 파일의 이름에서 '.'을 기준으로 나눠 해당 파일의 형식을 가져오고 파일의 크기는 byte, Kb, Mb단위로 나눠서 저장해주었다.
- 4) 해당 요소들을 sql에 넣고 수행을 해주는 것으로 업로드를 마무리했다.

구현 결과_파일 업로드_소스코드

```
int rowCount = 0;
String unum = request.getParameter("Unum");
String gnum = request.getParameter("Gnum");
PreparedStatement pstmt = null;
Statement stmt = null;
StringBuffer SQL = new StringBuffer("insert into file(F_name, F_form, F_size, F_update, F_last_f
SQL.append("values (?, ?, ?, ?, ?, ?, ?, ?)");
try {
    String encoding = "UTF-8";
    int sizeLimit = 10 * 1024 * 1024;
    String relativeDirectory = "/data/"+gnum;
    ServletContext context = getServletContext();
    String realDirectory = context.getRealPath(relativeDirectory);
    MultipartRequest multi = new MultipartRequest(request, realDirectory, sizeLimit, encoding, n

    Date now = new Date();
    SimpleDateFormat sf = new SimpleDateFormat("yyyy-MM-dd");
    String today = sf.format(now);
    String f_name = multi.getFileName("fileInput"); //파일이름
    String[] f_form = new String[2];
    f_form = f_name.split("[.]");
    String form = f_form[1];
    String f_size = String.valueOf(multi.getFile("fileInput").length()); // 파일의 크기를 반환
    int size = Integer.parseInt(f_size);

    if( (size/1024)>1){
        int size_kb = size/1024;
```

```
        if((size_kb/1024)>1){
            int size_mb = size_kb/1024;
            size_mb++;
            f_size= size_mb + "Mb";
        }else{
            size_kb++;
            f_size = size_kb+"Kb";
        }
    }else{
        f_size = f_size + "byte";
    }

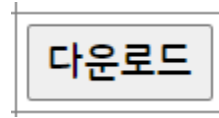
    pstmt = conn.prepareStatement(SQL.toString());
    //삽입할 학생 레코드 데이터 입력
    pstmt.setString(1, f_name);
    pstmt.setString(2, form);
    pstmt.setString(3, f_size);
    pstmt.setString(4, today);
    pstmt.setString(5, today);
    pstmt.setInt(6, Integer.parseInt(unum));
    pstmt.setInt(7, Integer.parseInt(gnum));
    pstmt.setString(8, relativeDirectory);

    rowCount = pstmt.executeUpdate();
    if (rowCount == 1) out.println("<hr>레코드 하나가 성공적으로 삽입 되었습니다.<hr>");
    else out.println("레코드 삽입에 문제가 있습니다.");

    stmt = conn.createStatement();
}
catch(Exception e) {
    out.println("조회에 문제가 있습니다. <hr>");
```

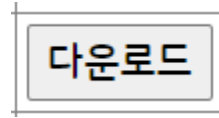


구현 결과_파일 다운로드



- 1) 다음 먼저 파일의 위치를 찾아야하기 때문에 해당 정보들을 DB에서 가져와서 파일의 실제 경로를 찾아준다.
- 2) 구한 경로를 통해 파일 객체를 생성하고 그 객체를 FileInputStream을 생성해서 바이트 단위로 읽어올 준비를 한다.
- 3) response헤더에 파일의 링크를 클릭했을 때 다운로드 화면이 출력되도록 처리를 해주고 파일의 크기도 같이 가져온다.
- 4) 그 다음 FileOutputStream에 response를 연결해서 연결했던 파일을 바이트단위로 가져오면서 파일의 다운로드가 완료된다.

구현 결과_파일 다운로드



- 1) 다음 먼저 파일의 위치를 찾아야하기 때문에 해당 정보들을 DB에서 가져와서 파일의 실제 경로를 찾아준다.
- 2) 구한 경로를 통해 파일 객체를 생성하고 그 객체를 FileInputStream을 생성해서 바이트 단위로 읽어올 준비를 한다.
- 3) response헤더에 파일의 링크를 클릭했을 때 다운로드 화면이 출력되도록 처리를 해주고 파일의 크기도 같이 가져온다.
- 4) 그 다음 FileOutputStream에 response를 연결해서 연결했던 파일을 바이트단위로 가져오면서 파일의 다운로드가 완료된다.

구현 결과_파일 다운로드_소스코드

```
try {

    pstmt = conn.prepareStatement(sql); // 해당 파일 정보 가져오기
    ResultSet result = pstmt.executeQuery();

    result.next();

    String f_name = result.getString("F_name");

    String relativeDirectory = result.getString("F_location");
    ServletContext context = getServletContext();
    String f_path = context.getRealPath(relativeDirectory); //경로 구하기
    f_path = f_path + "/" + f_name; //경로 끝

    try{
        myFile = new File (f_path); // 파일 인식
        f_in = new FileInputStream(myFile);
    }catch(FileNotFoundException fe){
        skip = true;
    }

    client = request.getHeader("User-Agent");

    // 파일 다운로드 헤더 지정
    response.reset() ;
    response.setContentType("application/octet-stream");
    response.setHeader("Content-Description", "JSP Generated Data");

    if(!skip){

        response.setHeader ("Content-Disposition", "attachment; filename="+
        response.setHeader ("Content-Length", ""+myFile.length() );

        f_os = response.getOutputStream();
        byte b[] = new byte[(int)myFile.length()];
        int leng = 0;
```

구현 결과_파일 삭제_소스코드

```
<%
String unum = request.getParameter("Unum");
String gnum = request.getParameter("Gnum");
String fno = request.getParameter("Fno");
PreparedStatement pstmt = null;
Statement stmt = null;
int rowCount = 0;

String sql_delete = "delete from file where F_num =" + fno;
String sql = "select * from file where F_num =" + fno;

try {

    pstmt = conn.prepareStatement(sql); // 해당 파일 정보 가져오기
    ResultSet result = pstmt.executeQuery();

    result.next();

    String relativeDirectory = result.getString("F_location");
    ServletContext context = getServletContext();
    String realDirectory = context.getRealPath(relativeDirectory); //경로 구하기

    realDirectory = realDirectory + "/" + result.getString("F_name");
    File deleteFile = new File (realDirectory);
    if(deleteFile.exists()){
        deleteFile.delete(); //파일이 존재하면 삭제
        //out.println("<hr>파일 삭제 성공.<hr>");
    }else{
        //out.println("<hr>파일 삭제 실패.<hr>");
    }

    pstmt = conn.prepareStatement(sql_delete);
    rowCount = pstmt.executeUpdate();
}
```

1)삭제는 해당하는 파일의 경로를 통해서 객체를 생성한다.

2)파일 객체를 .delete()함수를 사용하면 해당하는 파일이 삭제한다.

3)삭제가 되면 해당 파일의 DB정보도 sql명령문을 이용해 지워준다.

성과 및 기대효과

- 팀의 자료관리 서비스를 제공한다.
- 프론트와 백엔드 구현을 해봄으로 웹 이해도를 높인다.
- 팀 프로젝트를 진행함으로 협업의 중요성을 깨닫는다.
- 서버 세팅 단계에서부터 진행하여 서버에 대한 이해도를 높인다.

감사합니다