

식료품 쇼핑몰 사이트 개발 보고서



소 속	컴퓨터소프트웨어과
과 목 명	Backend Programming
담당교수	김진숙 교수
반	1반
팀 명	Org
팀 장	방정철
팀 원	김강현, 강지원
제 출 일	2023년 6월 21일



제목

팀명 : org

팀원 : 김강현, 강지원, 방정철

작성일 : 2023. 6. 21

개인 GITHUB 주소

<https://github.com/do04200611/BackEndTeamProject>

1. 개요

2. 구현 내용

3. 기대 효과


4. 참고 자료

목차

1. 팀원소개	5
2. 프로젝트개요	5
2.1 개요	
2.2 목적 및 필요성	
2.3 개발 일정	
3. 사이트 구현	7
3.1 개발 환경 및 도구	
가) 개발 환경	
나) 개발 도구	
3.2 구현 프로그램 목록	
3.3 데이터베이스 설계 및 구성	
3.4 인터페이스 및 기능 구현	
가) 메인	
나) 로그인	
다) 회원가입	
라) 게시판 글쓰기	
마) 게시판 글 수정/삭제하기 ...	
4. 기대효과 및 맺음말	16
5. 팀원 소감	16
6. 참고서적 및 참고 사이트	16
[첨부]	19
A. 프로젝트 계획서	
B. 발표자료	

C 주요 소스코드

1. 팀원 소개

팀원 이름(사진 포함)	담당역할	세부 수행 내용
김강현 	1.백엔드 및 프론트 엔드 코딩 2. 최종 보고서 및 발표 자료 작성	웹페이지 내 디자인 및 게시판과 회원가입이나 로그인 페이지내 DB를 연동하였습니다. 각오 : 아자 아자 파이팅!!!
방정철	백엔드 코드 코딩	DB를 연동하였습니다.
강지원	프론트 엔드 코드 코딩	웹 페이지 디자인

2. 프로젝트 개요

2.1 개요

2.2 목적 및 필요성

1. 흔히 쇼핑몰 사이트를 구현함으로써 향후에 많은 사람들이 온라인 쇼핑할 때 쓸 수 있습니다.
2. 많은 사람들이 사용하면 필요하거나 요긴하게 쓸 페이지를 만들 것입니다.
3. 코딩 작업을 통한 컴퓨터 프로그램에 관한 지식 습득을 할 수 있습니다.

2.3 개발일정

단계별 내용	시기		04월				05월		
	1주	2주	3주	4주	1주	2주	3주		
프로젝트 계획서 작성	→								
웹 페이지 디자인 설계	→	→							
웹 페이지 디자인 구현	→	→	→	→					
웹 페이지 기능 구현	→	→	→	→					
회원 가입 및 로그인 DB 연동			→	→	→	→	→	→	→
게시판 내 DB 연동				→	→	→	→	→	→
페이지 별 점검					→	→	→	→	→

3. 사이트 구현

3.1 개발 환경 및 도구

가) 개발 환경

▪ 시스템

구분	개발환경
서버	CPU : Intel(R) Xeon(TM) CPU 3.06GHz RAM : 3GB OS : Windows 2003 Server
클라이언트	CPU : Intel(R) Core(TM)2 Quad CPU Q6600 2.4GHz RAM : 2GB OS : Windows XP Sp2

▪ 웹사이트

구분	내용
URL(메인 홈페이지 URL)	http://localhost:8080/BackEndTeamProject/BackEndTeamProject/JSP/Html/HomePage.jsp
IP	203.237.244.91
Root Directory	/루트 디렉터리

나) 개발도구

구분	개발도구
Web Server	tomcat 9.0_40
Programming Language	Java1.8/JSP
Database	HediSQL
소스 편집	Eclipse
영상 편집	PhotoShop CS3
플래시 제작	Flash

구분	기능	파일명	설명	작성자	비고
2021학년도 1학기 웹 프로그래밍	0. 메인	HomePage.jsp	메인화면	김강현	
		Login.jsp	로그인 화면	김강현	membership 테이블에 login 데이터를 입력
JSP 파일		loginPro.jsp	로그인 처리	김강현	
	2. 회원가입	MemberManagementBulletinBoard.jsp	회원들 게시판	김강현	membership 테이블에 회원들 데이터를 등록
		MemberShipRegistration.jsp	회원 가입 화면	김강현	
		MemberShipRegistrationPro.jsp	회원 가입 처리 (등록)	김강현	
	3. 나의 정보(탈퇴)	MyInformation.jsp	회원 정보 수정 화면	김강현	membership 테이블에 회원들 데이터를 수정 및 삭제
		MyInformationPro.jsp	회원 정보 수정 처리	김강현	
		delete.jsp	회원 정보 삭제 처리	김강현	
	4. 응답 게시판 코드	RequestWritePro.jsp	회원들 문의사항 응답 게시판 작성 처리	김강현	request 테이블에 회원들 데이터를 수정 및 삭제 코드
		RequestWrite.jsp	회원들 문의사항 응답 게시판 작성 화면	김강현	
		RequestUpdatePro.jsp	회원들 문의사항 응답 게시판 수정 처리	김강현	
		RequestUpdateForm.jsp	회원들 문의사항 응답 게시판 수정 화면	김강현	
		RequestList.jsp	회원들 문의사항 응답 게시판 목록 화면	김강현	
		RequestDelete.jsp	회원들 문의사항 응답 게시판 글 삭제 처리	김강현	
		RequestContent.jsp	회원들 문의사항 응답 게시판 글 내용 화면	김강현	
	5. 자주 묻는 질문 게시판 코드	QuestionAndAnswerlogin.jsp	자주 묻는 질문 게시판 화면	김강현	자주 묻는 질문 게시판 코드(DB 연동 없음)
	6. 상품 후기 게시판(앨범 게시판)코드	ProductReviewWrite.jsp	상품 후기 게시판 화면	김강현	상품 후기 작성(DB 연동 코드)
		ProductReviewList.jsp	상품 후기 게시판 목록 화면	김강현	
		ProductReviewContent.jsp	상품 후기 게시판 내용 화면	김강현	
	7. 공지사항 게시판 코드	NoticeContent.jsp	공지사항 게시판 내용 화면	김강현	공지사항 게시판 코드(DB

CSS파일		NoticeList.jsp	공지사항 게시판 화면	김강현	연동 코드)
		NoticeWrite.jsp	공지사항 게시판 화면	김강현	
		NoticeDelete.jsp	공지사항 게시판 삭제 처리	김강현	
		NoticeWritePro.jsp	공지사항 게시판 작성 처리	김강현	
	1. 자주 묻는 질문 게시판 코드	FrequentlyAskedQuestionsBoard.css	자주 묻는 질문 게시판 css 코드	김강현	자주 묻는 질문 게시판 CSS 코드
		QuestionAndAnswer.css	자주 묻는 질문 게시판 css 코드	김강현	자주 묻는 질문 게시판 CSS 코드
	2. 공지사항 게시판 코드	NoticeContent.css	공지사항 게시판 내용 css 코드	김강현	공지사항 게시판 CSS 코드
		NoticeList.css	공지사항 게시판 목록 css 코드	김강현	
		NoticeContent.css	공지사항 게시판 내용 css 코드	김강현	
	3. 상품 후기 게시판 코드	Productreviewcontent.css	상품 후기 게시판 내용 css 코드	김강현	상품 후기 게시판 CSS 코드
		Productreviewlist.css	상품 후기 게시판 목록 css 코드	김강현	
		Productreviewwrite.css	상품 후기 게시판 작성 css 코드	김강현	
	4. 화면 꾸미기 css 코드	bootstrap.css	화면 꾸미기 css 코드	김강현	화면 꾸미기 css 코드
	5. 응답 게시판 코드	RequestContent.css	응답 게시판 내용 css 코드	김강현	응답 게시판 css 코드
		Request.css	응답 게시판 css 코드	김강현	
		RequestList.css	응답 게시판 목록 css 코드	김강현	
		RequestWrite.css	응답 게시판 작성 css 코드	김강현	
	7. 브랜드 마크 사진 css 코드	BrandMark	브랜드 마크 사진 css 코드	김강현	브랜드 마크 사진 css 코드
	8. 화면 꾸미기 css 코드	Custom.css	화면 꾸미기 css 코드	김강현	화면 꾸미기 css 코드

CSS파일	드				
	10.홈페이지 css 코드	HomePage.css	홈페이지 css 코드	김강현	홈페이지 css 코드
	11. 로그인 코드	Login.css	로그인 css 코드	김강현	로그인 css 코드
		LogoutHomePage.css	로그아웃 홈페이지 css 코드	김강현	
	12. 회원게시판 코드	MemberManagementBulletinBoard.css		김강현	
	14. 나의 정보 변경 css 코드	MyInformation.css	나의 정보 변경 css 코드	김강현	나의 정보 변경 css 코드
	16. 메뉴바 css 코드	UpponMenubar	상단 메뉴바 css 코드(게시판에 있는 블록형 메뉴)	김강현	메뉴바 css 코드
		VerticalMenubar	수직 메뉴바 css 코드	김강현	
		Menubar.css	메뉴바 css 코드	김강현	
이미지 파일 (png, jpg 파일 있음)	홈페이지 상품 이미지 파일	daesangwelllifegoods.jpg	홈페이지 내 있는 상품 이미지	김강현	홈페이지 내 있는 상품 이미지
		dongseorichisgoods.png	홈페이지 내 있는 상품 이미지	김강현	
		jwgoods.jpg	홈페이지 내 있는 상품 이미지	김강현	
		namyanggoods.png	홈페이지 내 있는 상품 이미지	김강현	
		nongsimgoods.png	홈페이지 내 있는 상품 이미지	김강현	
		othernongsimgoods.jpg	홈페이지 내 있는 상품 이미지	김강현	
		samyanggoods.png	홈페이지 내 있는 상품 이미지	김강현	
		sweetpotato.png	홈페이지 내 있는 상품 이미지	김강현	
		daeseonglifesciencegoods.png	상품 이미지	김강현	
		samyanggoods.png	상품 이미지	김강현	
	연관 이미지	email.png	이메일 연관 이미지	김강현	연관 이미지
		imageplus.png	연관 이미지	김강현	
		usergender.png	연관 이미지	김강현	
		userpassword.png	연관 이미지	김강현	
	창작된 브랜드	forest-brand-mark.	브랜드 이미지	김강현	브랜드 이미지

	드 이미지	png			
		textforestbrandmark.png	브랜드 이미지	김강현	
	실존하는 브랜드 마크 이미지	lottemark.png	실존하는 브랜드 마크 이미지	김강현	
DB설계 및 구현	테이블	org_방정철.sql	DB설계 및 구현	김강현	Request와 Membership 테이블을 생성한 파일입니다.
발표자료	PPT	org_방정철(발표자료).pptx	발표 자료	김강현	발표 자료
최종보고서 작성	HWP	org_방정철(최종 보고서)	최종 보고서	김강현	최종 보고서
프로젝트 계획서	HWP	org_방정철(프로젝트 계획서)	프로젝트 계획서	김강현	프로젝트 계획서
context.xml	xml	context.xml	context.xml	김강현	context.xml
라이브러리	라이브러리	commons-lang3-3.12.0-bin (1).tar.gz	라이브러리	김강현	라이브러리
		javax.servlet.jsp.jstl-api-1.2.2.jar	라이브러리	김강현	
		jstl-1.2.jar	라이브러리	김강현	
		mariadb-java-client-3.1.3 (1).jar	라이브러리	김강현	
		taglibs-standard-impl-1.2.5.jar	라이브러리	김강현	
DTO 파일	DTO 파일	BulletinBoardDto	DTO 파일	김강현	DTO 파일(게시판 DB 연동할 때 쓴 파일)
DAO 파일	DAO 파일	BulletinBoardDao	DAO 파일	김강현	DAO 파일(게시판 DB 연동할 때 쓴 파일)

3.2 구현 프로그램 및 제출 자료 작성 목록

3.3 데이터베이스 설계 및 구성

가) ERD

나) 테이블

[테이블명 : membership]

번호	컬럼 ID	컬럼 설명	형태	길이	NULL	비 고
1	userID	아이디	varchar	50		PK
2	userPassword	비밀번호	varchar	50	NOT NULL	
3	userEmail	이메일	varchar	50	NOT NULL	
4	userGender	성별	varchar	50	NOT NULL	

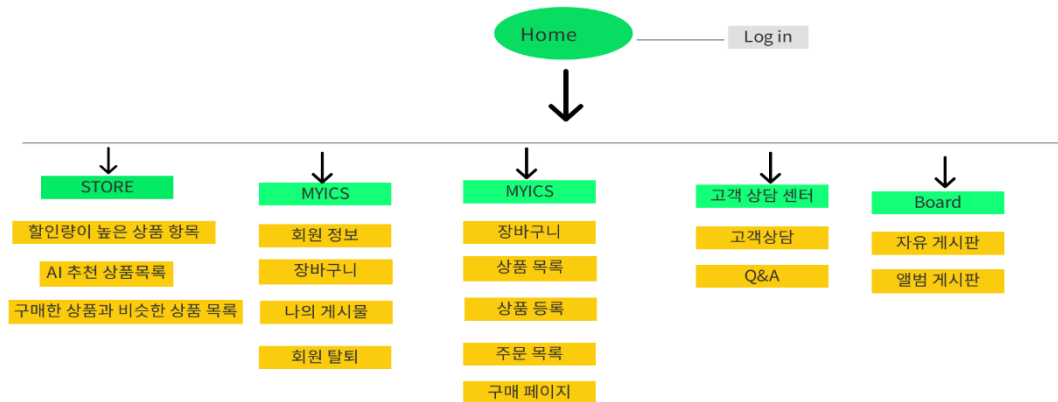
[테이블명 : request]

번호	컬럼 ID	컬럼 설명	형태	길이	NULL	비 고
1	ID	숫자	varchar	50	NOT NULL	
2	Title	제목	varchar	50	NOT NULL	
3	Writer	작성자	varchar	50	NOT NULL	
4	content	내용	varchar	50	NOT NULL	

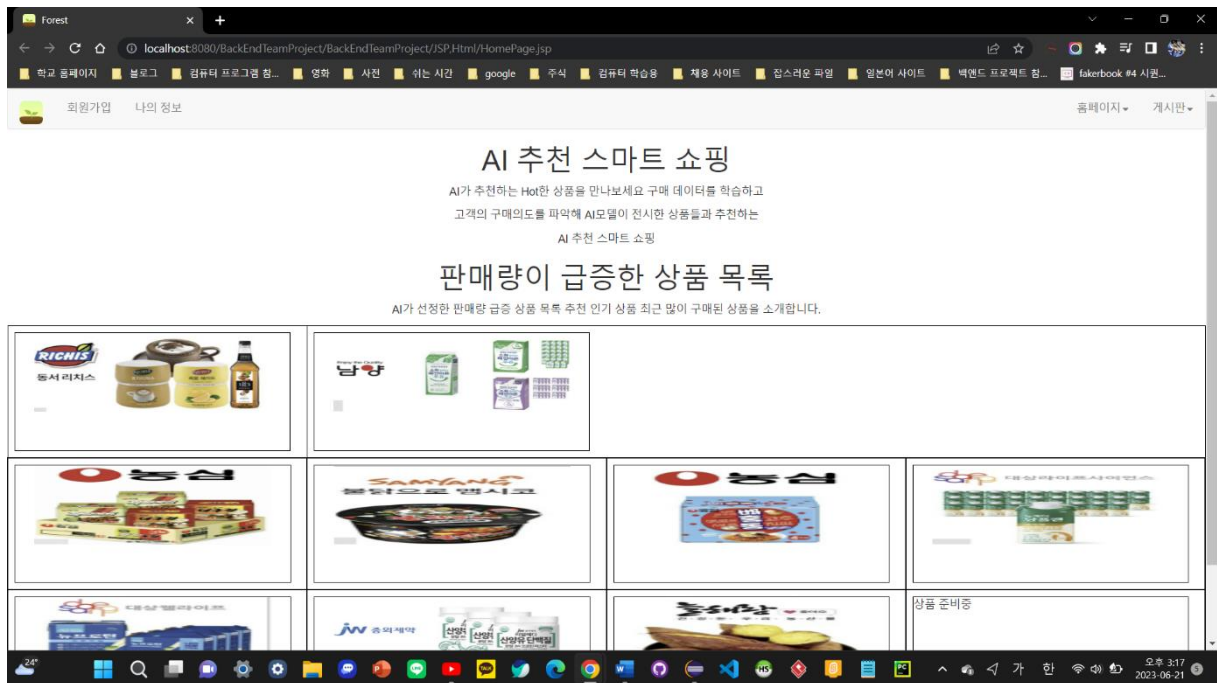
3.4 인터페이스 및 기능 구현

- 해당 웹사이트에서 구현된 기능들에 대한 상세히 설명한다(아래의 내용은 예제이므로 각 팀에게 적합하게 변경할 수 있다.)
- UML의 내용을 고려하여 작성한다.

가) 사이트 맵



나) 메인



화면명	로그인 입력	파일명	Login.jsp, loginPro.jsp
<p>① 입력사항을 입력 후 “확인”을 클릭하면 해당 사용자 확인작업 후 홈페이지로 이동한다. 입력한 정보가 틀린 경우는 아래와 같은 메시지 처리 후 로그인 화면으로 다시 이동한다.</p> <p>② 입력절차</p> <ul style="list-style-type: none"> - 해당 사용자 ID를 입력한다. - 비밀번호를 입력한다. 			

다) 로그인

forest x +

localhost:8080/BackEndTeamProject/BackEndTeamProject/JSP/Html/Login.jsp

학교 홈페이지 블로그 컴퓨터 프로그램 참... 영화 사진 쉬는 시간 google 주식 컴퓨터 학습용 채용 사이트 잡스러운 파일 일본어 사이트 백엔드 프로젝트 참... fakenbook #4 시원...

회원가입 나의 정보 홈페이지 계시판

로그인 화면

아이디


비밀번호

로그인


24°

오전 3:19
2023-06-21


라) 회원가입

 [회원가입](#) [나의 정보](#) [홈페이지](#) [게시판](#)


회원 가입 화면




아이디




비밀번호



비밀번호



이메일




성별

확인

취소

마) 게시판 글쓰기

 [회원가입](#) [게시판](#) [홈페이지](#) [게시판](#)

답변 게시판 글쓰기

글 제목

등록하기

작성취소

바) 게시판 글 수정/삭제하기

null

사용자 정보 변경

아이디:

null

제목:

null

내용:

null

작성자:

null

변경하기

작성취소

삭제

4. 기대효과 및 맺음말

많은 사람들이 홈쇼핑 웹 페이지를 이용함으로써 홈쇼핑의 시장과 온라인 업계 시장에 큰 획을 그을 수 있었다.

5. 팀원 소감

- 팀원 별로 지금까지의 애로점, 생각, 느낌, 각오, 계획에 대해 느낀 점을 A4용지의 반 페이지 이상씩 기술한다.

팀원 김강현 :

프로젝트를 계획하면서 정철이 형이 홈쇼핑 사이트를 하자고 하길래 알았다고 했지만, 저는 1학년 때 교내 대회 부트 캠프에서 프로젝트를 진행해서 그걸 토대로 코딩을 해서 페이지를 구성하고 싶었습니다. 하지만, 교수님이 추천하신 주제는 홈쇼핑 사이트라서 교수님이 추천하신 주제는 뭔가 이유가 있고 그 주제만이 가지고 있는 장점이 있을 것이라고 생각을 해서 군말없이 홈쇼핑 사이트를 주제로 결정하였습니다. 생각 했던 것보다 다른 주제보다 해야 할 것도(장바구니, 다수의 게시판 등) 많고 부족한 점이 많아서 어떻게 해야 하나 같은 고민을 했습니다. 그런 고민을 하다가 정철이 형의 질책이 제가 좀 더 빨리, 잘 할 수 있을 것 같다는 생각이 들었습니다. 웹페이지 디자인을 구현하는데 주로 쓰는 언어가 HTML과 CSS인데, 1학년 때 부트 캠프에서 디자인도 해왔고, 취업도 프론트 엔드 분야로 취업을 하고 싶습니다. 그렇기 때문에 HTML, CSS는 웹이나 프로그램 업계에서는 굉장히 쉬운 축에 속하는 언어라고 느꼈습니다. 그렇기 때문에, 큰 무리 없이 만들 수 있을 것이라고 생각했습니다. 그런데, 다른 페이지로 이동할 때 경로 지정이나, 이미지 삽입을 위해서 경로 설정을 하는데 큰 어려움을 느꼈고, 그런 어려움 속에서도 인터넷 속 자료를 찾고 해서 결국에는 이미지 삽입과 다른 페이지로 이동할 때 경로 지정까지 해내는 등의 성과를 이뤄냈습니다. 더 힘들었던 건 제가 손수 힘겹게 만든 웹 페이지 디자인이 다른 웹사이트를 찾아보니 좀 더 나아보여서 그 페이지 속 코드를 이용해서 디자인을 다시 구성 하는 경우가 몇 번 있었습니다. 그러면서 굉장한 허무함과 '아...내가 이러려고 코딩을 한 것이 아닌데' 같은 생각이 많이 들었습니다. JSTL, DAO DTO 같은 방식으로 학교에서는 DB 연동을 꽤 쉽게 해서 집에서는 당연히 프로젝트에 적용시킬 때는 꽤나 쉽게 될 것이라고 생각했는데, 생각 했던 것보다 말처럼 잘 되지 않아서 굉장한 스트레스를 느꼈습니다. 또, 그런 스트레스 속에서도 어떻게 해서라도 성과를 내고 싶어서 밤까지 새가면서 프로젝트에 참여했습니다. 소위 말하는 회사원이 많이 하는 야근을 한 기분이라서 굉장히 힘들었고, 회사 생활을 미리 한다고 생각하면서 후기를 쓰고 있습니다. 다음에도 이런 프로젝트가 있으면 스트레스를 느낄 까봐 겁도 나지만, 이런 뿌듯함을 느끼기 위해서 프로젝트에 참여할 생각입니다.

팀원 방정철 : 없음

팀원 강지원 : 없음

6. 참고서적 및 참고 사이트(예)

- <https://cooltext.com/> : 브랜드 마크 사진 만들 때 쓴 사이트
- <https://www.pngwing.com/ko/free-png-mvpuz/download> : 이미지 무료 다운로드 사이트
- <https://velog.io/@dpdnjs402/jnb845on> : 자주 묻는 질문게시판에 표 참조 코드

[첨부] : A. 프로젝트 계획서

백엔드프로그래밍 프로젝트 계획서

1. 팀원 정보

학과	컴퓨터소프트웨어과	작성일	2023년 04월 12일
과목	백엔드프로그래밍		
팀명	org		
팀장 및 팀원	성명	학번	주역할
	박정철	202080398	백엔드 구현
	김강현	202032291	전반적인 웹페이지 구성 전반적인 웹 페이지 내 DB 연동
	김지원	202236255	웹 페이지 디자인 구현

2. 과제 주제 및 목표

주제 : 식료품 쇼핑몰
 목표 : 게시판과 회원 가입 로그인할 때 DB 연동 완료하기

3. 요구사항 명세(정보, 디자인, 구현기능, 구현 우선순위 등 그림으로 그려내도 됨)

- 구현 기능: DB 연동할 한 로그인 및 회원가입
- 게시판 DB 코드 구현
- 게시판 수정 삭제 하기
- 종소형에 있는 페이지 구현하기

5. 레이아웃 설계 (메인)

The diagram illustrates the layout structure of the web application. It starts with a central 'Home' node, which branches into five main sections: 'Header', 'Main', 'Footer', 'Login', and 'Signup'. Each of these sections further branches into more specific components. For example, 'Header' branches into 'Navigation' and 'Search', while 'Main' branches into 'List', 'Form', and 'Page'. This hierarchical structure helps in organizing the layout and ensuring that all necessary components are included in the design.

Below the diagram is a screenshot of a web application interface. The interface shows a header with a navigation menu, a main content area with a list of products, and a footer. The products are displayed in a grid format, each with a product image, name, and price. The interface is clean and modern, with a white background and blue accents.

7. 참고자료 및 참고 웹사이트

책 JSP 웹 프로그래밍 (이성욱, 장종준 공저)

반드시 구현해야 할 기능

- 로그인/로그아웃
- 회원관리
- 게시판(다음 중 2개 이상)
 - 일반 게시판 / 자료 게시판 / 앨범 게시판 / 답변 게시판
- 각 코드마다 주석 달기

[참고] 사이트 맵의 트리 형태

2023학년도 1학기

발표 평가표

구분	내용	점수
발표내용	주제에 대한 명확한 설명이 되었는가?	50
	슬라이드 자료가 발표 목적에 적절히 작성되었는가?	
	발표자가 내용을 정확하게 이해하고 발표하였는가?	
발표기술	발표시간을 준수하였는가?	20
	쉬운 표현으로 청중이 이해할 수 있도록 발표하였는가?	
발표태도	절제하게 발표준비를 하였는가?	30
	진지한 태도(의상, 표정)로 발표하였는가?	
	청중이 지루하지 않게 물임을 유도하였는가?	
	합계	100

웹사이트 구축 평가표

번호	내용	점수
1	전체 웹사이트의 기능이 목적에 맞게 설계되고 구현되어 있는가?	10
2	웹사이트의 기능을 사용하는데 어려움 없이 직관적이고 편리한가?	10
3	로그인/로그아웃 기능이 구현되어 제대로 기능하고 있는가?	15
4	게시판의 CRUD 기능이 구현되어 제대로 기능하고 있는가?	20
5	2개의 게시판이 서로 다른 목적으로 구현되어 있는가?	5
6	기본적으로 구현되어야 하는 기능 이외에 기능이 추가되어 있는가(예: 관리자 기능 등)?	5
7	각 파일마다 적절한 주석이 작성되어 있는가?	10
8	데이터 테이블이 적절하게 설계되어 사용할 수 있는가?	10
9	최종 보고서에서 각 항목이 빠짐없이 정리되어 있는가?	10
10	최종보고서의 전체 내용을 충실하게 구성하여 정리하였는가?	5
	합계	100


[첨부] : B. 발표 자료

백엔드 프로젝트 발표 자료

팀원 : 방정철, 김강현, 강지원



목차

1. DB 코드
 2. DB 코드 설명
 3. 회원 관리 게시판 UI 및 코드 설명
 4. 메인 페이지 및 대표 UI
 5. 자주 묻는 질문 게시판 UI
 6. 자주 묻는 질문 게시판 코드 설명
- 

DB 코드

```
CREATE TABLE `request` (  
  `id` VARCHAR(50) NULL DEFAULT NULL COLLATE 'utf8_general_ci',  
  `title` VARCHAR(50) NULL DEFAULT NULL COLLATE 'utf8_general_ci',  
  `writer` VARCHAR(50) NULL DEFAULT NULL COLLATE 'utf8_general_ci',  
  `content` VARCHAR(50) NULL DEFAULT NULL COLLATE 'utf8_general_ci'  
)  
COLLATE='utf8_general_ci'  
ENGINE=InnoDB
```

다음 코드는 DB 테이블을 만들 때 사용한 코드입니다.

```
CREATE TABLE `membership` (  
  `userID` CHAR(50) NOT NULL DEFAULT '' COLLATE 'utf8_general_ci',  
  `userPassword` CHAR(50) NULL DEFAULT NULL COLLATE 'utf8_general_ci',  
  `userEmail` CHAR(50) NULL DEFAULT NULL COLLATE 'utf8_general_ci',  
  `userGender` CHAR(50) NULL DEFAULT NULL COLLATE 'utf8_general_ci',  
  PRIMARY KEY (`userID`) USING BTREE  
)  
COLLATE='utf8_general_ci'  
ENGINE=InnoDB
```

DB 코드 설명

테이블은 두개를 만들었으며,
request, membership 테이블을
만들었고, 회원 가입 및 게시판
등록 삭제 수정 하는데 사용할
테이블입니다.

회원 관리 게시판 UI 및 코드 설명

```
<tr>
  <td>
    <a href="MyInformation.jsp?userID=<%=userID%>"><%=userID%></a>
  </td>
  <td><%=userPassword%></td>
  <td><%=userEmail%></td>
  <td><%=userGender%></td>
</tr>
```

아이디	비밀번호	이메일	성별
1	123	123	321
11naver@naver.com	11naver12	11naver@naver.com	남
12	12	12	1@n.com
13	11	1@n.com	남
132			
kak	kg@n.co	대성	1@n.co
kak 1	1	11naver@naver.com	대성
000	888	11@n.com	여자

회원 관리 게시판이고 회원들의 정보를 볼 수 있습니다.

메인 페이지 및 대표 UI



로그인 화면

로그인

자주 묻는 질문 게시판 UI



자주 묻는 질문 게시판 코드 설명

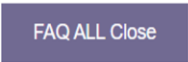
```
window.onload = () => {  
  const panelFaqContainer = document.querySelectorAll(".panel-faq-container");  
  
  let panelFaqAnswer = document.querySelectorAll(".panel-faq-answer");  
  
  const btnAllClose = document.querySelector("#btn-all-close");  
  
  for( let i=0; i < panelFaqContainer.length; i++ ) {  
    panelFaqContainer[i].addEventListener('click', function() {  
      panelFaqAnswer[i].classList.toggle('active');  
    });  
  };  
};
```

사진 속 표를 누르면 고객들이 자주 물을 질문들을 목록으로 보여줍니다.


자주 묻는 질문 게시판 코드 설명

```
btnAllClose.addEventListener('click', function() {  
    for(let i=0; i < panelFaqAnswer.length; i++) {  
        panelFaqAnswer[i].classList.remove('active');
```

위의 코드는 아래 사진 속 FAQ ALL Close를 누르면 보고 있던 표가 닫히는 코드입니다.



FAQ ALL Close



저희들의 발표를 들어 주셔서
감사합니다

[첨부] : C. 주요 소스 코드

BulletinBoardDao 파일 코드

```
package cs.dit;

import java.sql.Statement;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ArrayList;
import javax.naming.Context;
import javax.naming.InitialContext;
import javax.sql.DataSource;

public class BulletinBoardDao {

    private Connection getConnection() throws Exception {
        // 데이터베이스 연결을 위한 Connection 객체를 반환하는 메서드입니다.
        Context initCtx = new InitialContext();
        Context envCtx = (Context) initCtx.lookup("java:comp/env");
        DataSource ds = (DataSource) envCtx.lookup("jdbc/gh");
        Connection con = ds.getConnection();
        return con;
    }

    public void insert(BulletinBoardDto dto) {
        // 게시글을 데이터베이스에 삽입하는 메서드입니다.
        String sql = "INSERT INTO request(ID, Title, Content, Writer ) VALUES(?,?,?,?)";
        try (
            Connection con = getConnection();
            PreparedStatement pstmt = con.prepareStatement(sql);
```

```
        )

    {

        pstmt.setString(1, dto.getId());
        pstmt.setString(2, dto.getTitle());
        pstmt.setString(3, dto.getContent());
        pstmt.setString(4, dto.getWriter());

        pstmt.executeUpdate();
    } catch (Exception e){
        e.printStackTrace();
    }
}

public ArrayList<BulletinBoardDto> list() {
    // 게시글 목록을 데이터베이스에서 가져오는 메서드입니다.
    String sql = "SELECT * FROM request";
    ArrayList<BulletinBoardDto> dtos = new ArrayList<BulletinBoardDto>();

    try (Connection con = getConnection();
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery(sql);
        )
    {

        while (rs.next()) {
            BulletinBoardDto dto = new BulletinBoardDto();

            dto.setId(rs.getString("id"));
            dto.setTitle(rs.getString("title"));
            dto.getWriter(rs.getString("writer"));
```

```
        dto.setContent(rs.getString("content"));
        dtos.add(dto);
    }
} catch (Exception e) {
    e.printStackTrace();
}

return dtos;
}

public BulletinBoardDto selectOne(String id) {
    // 특정 게시글을 데이터베이스에서 조회하는 메서드입니다.
    String sql = "SELECT * FROM request WHERE id=?";
    BulletinBoardDto dto = new BulletinBoardDto();

    try (Connection con = getConnection();
        PreparedStatement pstmt = con.prepareStatement(sql);
    )
    {
        pstmt.setString(1, id);

        try (ResultSet rs = pstmt.executeQuery();)
        {
            rs.next();

            dto.setId(id);
            dto.setWriter(rs.getString("writer"));
            dto.setContent(rs.getString("content"));
            dto.setTitle(rs.getString("title"));

        } catch (Exception e) {
            e.printStackTrace();
        }
    } catch (Exception e) {
```

```
        e.printStackTrace();
    }

    return dto;
}

public void update(BulletinBoardDto dto) {
    // 게시글을 수정하는 메서드입니다.
    String sql = "UPDATE request SET title =?, writer=?,content= ? WHERE Id
=?";

    try (
        Connection con = getConnection();
        PreparedStatement pstmt = con.prepareStatement(sql);
    )
    {
        pstmt.setString(1, dto.getTitle());
        pstmt.setString(2, dto.getWriter());
        pstmt.setString(3, dto.getContent());
        pstmt.setString(4, dto.getId());

        pstmt.executeUpdate();
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }

}

public void delete(String id) {
    // 게시글을 삭제하는 메서드입니다.
    String sql = "DELETE FROM request WHERE id =?";

    try(
```



```
        Connection con = getConnection();
        PreparedStatement pstmt = con.prepareStatement(sql);
    )
    {
        pstmt.setString(1, id);
        pstmt.executeUpdate();

    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
}
/*
 * public void loginChange(BulletinBoardDto dto, String flag) {
 * PreparedStatement pstmt = null; try (Connection con = getConnection(); ) { if
 * (flag.equals("i")) { String sql =
 * "INSERT INTO request (Content, Title, WRITER) VALUES(?, ?, ?)"; pstmt =
 * con.prepareStatement(sql); pstmt.setString(1, dto.getContent());
 * pstmt.setString(2, dto.getTitle()); pstmt.setString(3, dto.getWriter()); }
 *
 * } }
 */
}
```

Login.jsp 파일 코드

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>
```

```
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">

<!--뷰포트-->

<meta name="viewport" content="width=device-width"
initial-scale="1">

<!-- 스타일시트-->

<link rel="stylesheet" href="../CSS/MenuBar.css">

<link rel="stylesheet" href="../CSS/bootstrap.css">

<link rel="stylesheet" href="../CSS/custom.css">

<link rel="stylesheet" href="../CSS/Login.css">

<link rel="stylesheet" href="../CSS/BrandMark.css">

<link rel="shortcut icon" type="image/x-icon"
href="../../images/forest-brand-mark.png">

<title>forest</title>

</head>

<body>

<!-- 네비게이션-->

<nav class="navbar navbar-default">

<div class="navbar-header">

<button type="button" class="navbar-toggle collapsed"
data-toggle="collapse" data-target="#bs-example-navbar-
collapse-1"
aria-expanded="false">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>
```

```
</button>

<a class="navbar-brand" href="HomePage.jsp">



</a>

</div>

<div class="collapse navbar-collapse" id="bs-example-navbar-
collapse-1">

<ul class="nav navbar-nav ">

<li><a href="MemberShipRegistration.jsp">회원가입</a></li>

<li><a href="MyInformation.jsp">나의 정보</a></li>

</ul>

<ul class="nav navbar-nav navbar-right">

<li class="dropdown">

<a href="Request/RequestList.jsp" class="dropdown-toggle"
data-toggle="dropdown" role="button" aria-haspopup="true"
aria-expanded="false">게시판<span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="productreview/ProductReviewList.jsp">상품
후기</a></li>

<li><a href="QuestionAndAnswer.jsp">자주 묻는 질문</a></li>

<li><a href="Request/RequestList.jsp">답변 게시판</a></li>

<li><a href="notice/NoticeList.jsp">공지 사항</a></li>

</ul>

</li>
```

```
</ul>

<ul class="nav navbar-nav navbar-right">

<li class="dropdown">

<a href="HomePage.jsp" class="dropdown-toggle"
data-toggle="dropdown" role="button" aria-haspopup="true"
aria-expanded="false">홈페이지<span class="caret"></span></a>

<ul class="dropdown-menu">

<li class="active"><a href="Login.jsp">로그인</a></li>

</ul>

</li>

</ul>

</div>

</nav>

<!-- 로그인폼-->

<div class="container">

<div class="col-lg-4"></div>

<div class="col-lg-4">

<!-- 정보트론-->

<!-- 로그인 정보 숨기고 전송-->

<h3 style="text-align: center;">로그인 화면</h3>

<form action="loginPro.jsp" method="post">

<div class="form-group">

<input type="image" src="../../../images/email.png"
id="userIdimage">
```

```
<input type="text" class="form-control" placeholder="0+0/0/"
name="userID" id="userID" maxlength="20">

</div>

<div class="form-group">

<input type="image" src="../../images/userpassword.png"
id="userPasswordimage" >

<input type="password" class="form-control"
placeholder="비밀번호" name="userPassword" id="userPassword"
maxlength="20">

</div>

<input type="submit" class="btn btn-primary form-control"
value="로그인" onclick="location.href='loginPro.jsp.jsp'">

</form>

</div>

<div class="col-lg-4"></div>

</div>

<script src="https://code.jquery.com/jquery-
3.1.1.min.js"></script>

<script src="js/bootstrap.js"></script>

</body>

</html>
```