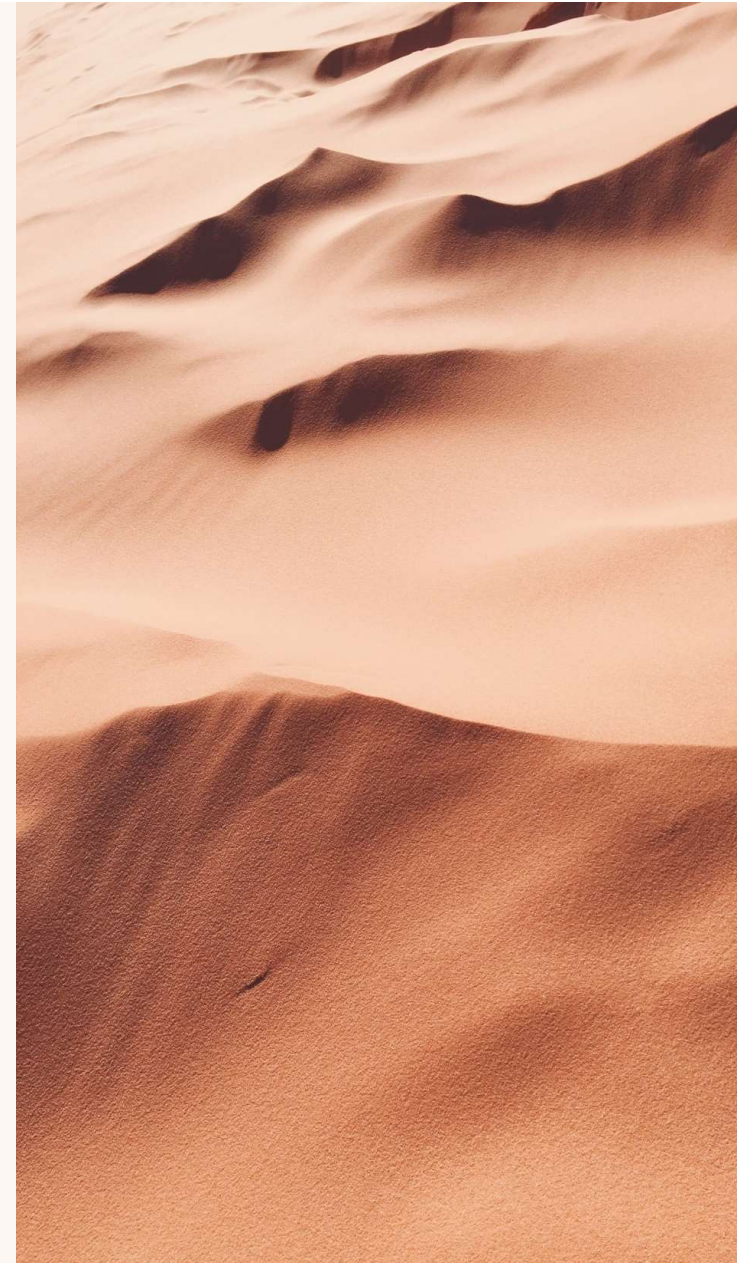


화면 구현

서정현

목차 A table of contents.

- 1.문제 1 - 프로젝트 구성
- 2.문제 2 - 게시판 목록
- 3.문제 3 - 회원정보 비밀번호 인증
- 4.문제 4 - 회원정보 설정



1. 문제 1 – 프로젝트 구성

1. 프로젝트명 : Jboard2
2. 데이터베이스 : java_board
3. 개발 도구 : java2, JSP, Tomcat9, MySQL8
4. 작업 내용 : 회원 정보수정 기능 추가하기
5. 제출 내용
 - 추가된 Controller, View, Model(Service, DAO) 코드 및 설명
 - 기능 동작 테스트 및 시연 화면

2. 문제 2 – 게시판 목록

1. 게시판 목록 상단 헤더 회원정보 링크 추가

Board System v2.0 허니님 반갑습니다. [회원정보](#) [로그아웃](#)

글목록

제목 ▼ 제목 키워드, 글쓴이 검색

번호	제목	글쓴이	날짜	조회
----	----	-----	----	----

```
<body>
  <div id="wrapper">
    <header>
      <h3>Board System v2.0</h3>
      <p>
        <span>${sessUser.nick}</span>님 반갑습니다.
        <a href='<c:url value="/user/info.do?uid=${sessUser.uid}"/>'>[회원정보]</a>
        <a href='<c:url value="/user/logout.do?uid=${sessUser.uid}"/>'>[로그아웃]</a>
      </p>
    </header>
  </div>
```

3. 문제 3 – 회원정보 비밀번호 인증

2. 비밀번호 입력 후 다음 버튼 클릭(AJAX 이용하여 회원인증) - Script

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" session="false"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<jsp:include page="._header.jsp"/>
<script>
    const contextRoot = location.pathname.split('/')[1]; // 컨텍스트루트 ex) Jboard2
    const uid = '${reqUser.uid}';
    $(function () {
        $('.btnNext').click(function () {
            const pass = $('input[name=pass]').val();
            console.log(pass);
            let jsonData = {
                "uid": uid,
                "pass": pass
            }
            $.ajax({
                url: "/" + contextRoot + "/user/info.do",
                method: 'post',
                data: jsonData,
                dataType: 'json',
                success: function (data) {
                    if (data.result == 1) {
                        location.href = "/" + contextRoot + "/user/myInfo.do"
                    } else {
                        alert('비밀번호가 일치하지 않습니다. ');
                    }
                }
            })
        })
    })
</script>
```

3. 문제 3 – 회원정보 비밀번호 인증

3. 비밀번호 입력 후 다음 버튼 클릭(AJAX 이용하여 회원인증) - InfoController.java

```
@Override
protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    logger.info("InfoController doGet...");
    req.getRequestDispatcher("/user/info.jsp").forward(req, resp);
}

@Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    String uid = req.getParameter("uid");
    String pass = req.getParameter("pass");

    UserVO vo = null;
    vo = service.selectUser(uid, pass);
    int result = 0;

    if(vo != null) {
        result = 1;
    }

    JsonObject json = new JsonObject();
    json.addProperty("result", result);
    resp.setContentType("json/application;charset=UTF-8");
    PrintWriter writer = resp.getWriter();
    writer.print(json.toString());
}
```

3. 문제 3 – 회원정보 비밀번호 인증

3. 비밀번호 입력 후 다음 버튼 클릭(AJAX 이용하여 회원인증) - UserService 및 UserDao

```
// 아이디 / 패스워드 확인
public UserVO selectUser(String uid, String pass) {
    return dao.selectUser(uid, pass);
}
```

※ 회원 정보를 조회한후 조건에 해당하는 회원이 있을 경우 회원 설정 화면으로 이동

```
// 자동 로그인 시 회원정보 가져오는 메서드
public UserVO selectUserBySessId(String sessId) {
    UserVO vo = null;

    try {
        logger.info("selectUser...");
        con = getConnection();
        psmt = con.prepareStatement(Sql.SELECT_USER_BY_SESSID);
        psmt.setString(1, sessId);

        rs = psmt.executeQuery();

        if(rs.next()) {
            vo = new UserVO();
            vo.setUid(rs.getString(1));
            vo.setPass(rs.getString(2));
            vo.setName(rs.getString(3));
            vo.setNick(rs.getString(4));
            vo.setEmail(rs.getString(5));
            vo.setHp(rs.getString(6));
            vo.setGrade(rs.getInt(7));
            vo.setZip(rs.getString(8));
            vo.setAddr1(rs.getString(9));
            vo.setAddr2(rs.getString(10));
            vo.setRegip(rs.getString(11));
            vo.setRdate(rs.getString(12));
        }

        close();
    } catch (Exception e) {
        logger.error(e.getMessage());
    }
    logger.debug("vo : " + vo);
    return vo;
}
```


문제 4 – 회원정보 설정

1. 기본적인 사용자 정보 출력하기 - myInfo.jsp 및 myInfoController doGet

회원정보 설정	
아이디	sih940520
비밀번호	<input type="password" value="비밀번호 입력"/>
비밀번호 확인	<input type="password" value="비밀번호 입력 확인"/> <input type="button" value="비밀번호 수정"/>
회원가입일	2022-10-21 16:50:31

개인정보 수정	
이름	<input type="text" value="서정현"/>
별명	<input type="text" value="공백없는 한글, 영문, 숫자 입력"/> <input type="button" value="중복확인"/>
이메일	<input type="text" value="ooo33425@naver.com"/> <input type="button" value="인증번호 받기"/>
휴대폰	<input type="text" value="010-3140-9982"/> <input type="button" value="인증번호 받기"/>
주소	<input type="text" value="61421"/> <input type="button" value="우편번호찾기"/> <input type="text" value="광주 동구 계림동 538-38"/> <input type="text" value="202호"/>
회원탈퇴	<input type="button" value="탈퇴하기"/>

```
@Override
protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    logger.info("MyInfoController doGet...");
    req.getRequestDispatcher("/user/myInfo.jsp").forward(req, resp);
}
```

```
<form action='<c:url value="/user/myInfo.do"/>' method="post">
  <table border="1">
    <caption>회원정보 설정</caption>
    <tr>
      <td>아이디</td>
      <td>${reqUser.uid}</td>
    </tr>
    <tr>
      <td>비밀번호</td>
      <td><input type="password" name="pass1" placeholder="비밀번호 입력"/></td>
    </tr>
    <tr>
      <td>비밀번호 확인</td>
      <td>
        <input type="password" name="pass2" placeholder="비밀번호 입력 확인"/>
        <button type="button" class='passUpdate'>비밀번호 수정</button>
        <br/><span class="passResult"></span>
      </td>
    </tr>
    <tr>
      <td>회원가입일</td>
      <td>${reqUser.rdate}</td>
    </tr>
  </table>
```


문제 4 – 회원정보 설정

2. 별명 중복 체크, 이메일 인증, 우편번호 검색기능 구현하기

회원정보 설정

아이디	sjh340520		
비밀번호	<input type="password"/>		
비밀번호 확인	<input type="password"/>	<input type="button" value="비밀번호 수정"/>	
회원가입일	2022-10-21 16:50:31		

개인정보 수정

이름	<input type="text" value="서정현"/>		
별명	<div>공백없는 한글, 영문, 숫자 입력</div> <div><input type="text" value="허니2"/> <input type="button" value="중복확인"/> 사용 가능한 별명입니다.</div>		
이메일	<div><input type="text" value="ooo33445@naver.com"/> <input type="button" value="인증번호 받기"/></div> <div><input type="text" value="인증번호 입력"/> <input type="button" value="확인"/></div> <div>인증코드를 전송했습니다. 이메일을 확인 하세요.</div>		
휴대폰	<div><input type="text" value="010-3140-9982"/> <input type="button" value="인증번호 받기"/></div> <div>이미 사용중인 휴대폰입니다.</div>		
주소	<div><input type="text" value="61421"/> <input type="button" value="우편번호찾기"/></div> <div><input type="text" value="광주 동구 계림동 538-38"/></div> <div><input type="text" value="202호"/></div>		
회원탈퇴	<input type="button" value="탈퇴하기"/>		

Daum Postcode Service - Chrome

about:blank

tip

아래와 같은 조합으로 검색을 하시면 더욱 정확한 결과가 검색됩니다.

도로명 + 건물번호

예) 판교역로 235, 제주 첨단로 242

지역명(동/리) + 번지

예) 삼성동 681, 제주 영평동 2181

지역명(동/리) + 건물명(아파트명)

예) 분당 주공, 연수동 주공3차

사서함명 + 번호

예) 분당우체국사서함 1~100

Powered by kakao | 우편번호 서비스 안내

```
<script src='<c:url value='/js/validation.js'/>'></script>
```

※ 기존에 회원가입에 이용한 validation.js를 활용

문제 4 – 회원정보 설정

3. 탈퇴하기 기능 구현

- 탈퇴시 회원 grade = 0 으로 수정
- 탈퇴 날짜 입력받기(탈퇴 날짜 컬럼 : wdate 추가)
- 탈퇴 완료 후 로그인으로 이동

```
$('#unRegister').click(function (e) {  
    e.preventDefault();  
    if(!confirm('정말 탈퇴 하시겠습니까?')){  
        return;  
    }  
    let jsonData = {  
        "uid":uid,  
    }  
    $.ajax({  
        url: "/" + contextRoot + "/user/unregister.do",  
        method: 'post',  
        data : jsonData,  
        dataType: 'json',  
        success: function (data) {  
            if(data.result == 1){  
                alert("회원탈퇴가 정상적으로 완료되었습니다.");  
                location.href = "/" + contextRoot + "/user/logout.do";  
            } else {  
                alert("회원탈퇴에 실패했습니다.")  
            }  
        }  
    })  
})
```

6	hp	CHAR	13	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	기본값 없음
7	grade	TINYINT	3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	'2'
8	zip	CHAR	5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
9	addr1	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
10	addr2	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
11	regip	VARCHAR	100	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	기본값 없음
12	rdate	DATETIME		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	기본값 없음
13	sessId	VARCHAR	100	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
14	sessLimitDate	DATETIME		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
15	wdate	DATETIME		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL

※ 탈퇴하기 버튼을 누르면 회원 아이디를 통해 해당 회원의 정보를 수정하는 AJAX

성공 시 : logout을 한 후에 login화면을 보여줌.

실패 시 : 회원탈퇴에 실패에 대한 안내를 함.

문제 4 – 회원정보 설정

3. 탈퇴하기 기능 구현

- unregisterController.java

```
@Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    logger.info("UnRegisterController...");
    String uid = ((UserVO)req.getAttribute("reqUser")).getUid();
    int result = service.deleteUser(uid);

    resp.setContentType("json/application;charset=UTF-8");
    PrintWriter writer = resp.getWriter();
    JsonObject json = new JsonObject();
    json.addProperty("result", result);
    writer.print(json.toString());
}
```

- UserService.java

```
// 회원 탈퇴
public int deleteUser(String uid) {
    return dao.deleteUser(uid);
}
```

문제 4 – 회원정보 설정

3. 탈퇴하기 기능 구현

- UserDao.java

```
// 회원탈퇴
public int deleteUser(String uid) {
    int result = 0;
    try {
        logger.info("deleteUser...");
        con = getConnection();
        pstmt = con.prepareStatement(Sql.DELETE_USER);
        pstmt.setString(1, uid);

        result = pstmt.executeUpdate();

        close();
    } catch (Exception e) {
        logger.error(e.getMessage());
    }
    logger.debug("result : " + result);
    return result;
}
```

```
// 회원정보 수정
public static final String DELETE_USER = "UPDATE `board_user` SET"
    + " `grade` = 0,"
    + " `wdate` = NOW()"
    + " WHERE `uid` = ?";
```

문제 4 – 회원정보 설정

3. 탈퇴하기 기능 구현 – 시연 화면

localhost 내용:
정말 탈퇴 하시겠습니까?

확인 취소

회원정보 설정

아이디	sjh940520		
비밀번호	비밀번호 입력		
비밀번호 확인	비밀번호 입력 확인	비밀번호 수정	
회원가입일	2022-10-21 16:50:31		

개인정보 수정

이름	서정현		
별명	공백없는 한글, 영문, 숫자 입력 허니 중복확인		
이메일	ooo33425@naver.com	인증번호 받기	
휴대폰	010-3140-9982	인증번호 받기	
주소	61421	우편번호 찾기	
	광주 동구 계림동 538-38 202호		
회원탈퇴	탈퇴하기		



localhost 내용:
회원탈퇴가 정상적으로 완료되었습니다.

확인

회원정보 설정

아이디	sjh940520		
비밀번호	비밀번호 입력		
비밀번호 확인	비밀번호 입력 확인	비밀번호 수정	
회원가입일	2022-10-21 16:50:31		

개인정보 수정

이름	서정현		
별명	공백없는 한글, 영문, 숫자 입력 허니 중복확인		
이메일	ooo33425@naver.com	인증번호 받기	
휴대폰	010-3140-9982	인증번호 받기	
주소	61421	우편번호 찾기	
	광주 동구 계림동 538-38 202호		
회원탈퇴	탈퇴하기		



Board System v2.0

로그인

아이디 입력
비밀번호 입력
자율 로그인

회원 로그인 안내
아래 회원이 아니시면 회원으로 가입하세요.
아이디 찾기 | 비밀번호 찾기 | 회원가입

©Copyright seo.or.kr

name	nick	email	hp	grade	zip	addr1	addr2	regip	rdate	sessId	sessLimitDate	wdate
서정현	허니	ooo33425@naver.com	010-3140-9982	0	61421	광주 동구 계림동 538-38	202호	0:0:0:0:0:0:1	2022-10-21 16:50:31	(NULL)	(NULL)	2022-12-05 17:50:29

문제 4 – 회원정보 설정

3. 회원 수정하기 기능 구현

- 비밀번호를 입력하지 않으면 기존 비밀번호 유지
- 이름, 별명, 이메일, 휴대폰, 주소 수정하기
- 회원수정 후 게시판 목록으로 이동

```
const uid = '${reqUser.uid}';
$(function () {
    isUidok = true;
    isPassok = true;
    isNameok = true;
    isNickok = true;
    isEmailok = true;
    isEmailAuthok = true;
    isHpok = true;
    $('.passUpdate').click(function () {
        const pass = $('input[name=pass2]').val();
        let jsonData = {
            "uid":uid,
            "pass1":pass
        }

        $.ajax({
            url: "/" + contextRoot + "/user/findPwChange.do",
            method: 'post',
            data : jsonData,
            dataType: 'json',
            success: function (data) {
                if(data.result == 1){
                    $('.passResult').val("비밀번호가 수정되었습니다.")
                } else {
                    $('.passResult').val("비밀번호가 수정에 실패했습니다.")
                }
            }
        })
    })
})
```

※ 비밀번호 수정 버튼을 누르면 실행되는 AJAX

문제 4 – 회원정보 설정

3. 회원 수정하기 기능 구현

- 이름, 별명, 이메일, 휴대폰, 주소 수정하기

```
<form action='<c:url value="/user/myInfo.do"/>' method="post">
@Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    logger.info("MyInfoController doPost...");
    String uid = ((UserVO)req.getAttribute("reqUser")).getUid();
    String name = req.getParameter("name");
    String nick = req.getParameter("nick");
    String email = req.getParameter("email");
    String hp = req.getParameter("hp");
    String zip = req.getParameter("zip").equals("") ? null : req.getParameter("zip");
    String addr1 = req.getParameter("addr1").equals("") ? null : req.getParameter("addr1");
    String addr2 = req.getParameter("addr2").equals("") ? null : req.getParameter("addr2");

    // 수정된 회원 정보 세션에 수정
    HttpSession session = req.getSession(false);
    UserVO sessUser = (UserVO)session.getAttribute("sessUser");
    sessUser.setNick(nick);
    sessUser.setName(name);
    sessUser.setEmail(email);
    session.setAttribute("sessUser", sessUser);

    UserVO vo = new UserVO();
    vo.setUid(uid);
    vo.setName(name);
    vo.setNick(nick);
    vo.setEmail(email);
    vo.setHp(hp);
    vo.setZip(zip);
    vo.setAddr1(addr1);
    vo.setAddr2(addr2);

    int result = 0;
    result = service.updateUserInfo(vo);

    if(result == 1) {
        resp.sendRedirect(req.getContextPath() + "/list.do");
    } else {
        JSFunction.alertBack(resp, "회원 수정에 실패했습니다.");
    }
}
```

※ 회원수정 버튼을 누르면 myInfoController로 넘어오면서 수정을 진행한다.

문제 4 – 회원정보 설정

3. 회원 수정하기 기능 구현

- 이름, 별명, 이메일, 휴대폰, 주소 수정하기 – Userservice.java 및 UserDAO.java

```
/** myInfo */  
// 회원 정보수정  
public int updateUserInfo(UserVO vo) {  
    return dao.updateUserInfo(vo);  
}
```

```
/** myInfo */  
// 회원정보 수정  
public int updateUserInfo(UserVO vo) {  
    int result = 0;  
    try {  
        logger.info("updateUserInfo...");  
        con = getConnection();  
        pstmt = con.prepareStatement(Sql.UPDATE_USER_INFO);  
        pstmt.setString(1, vo.getName());  
        pstmt.setString(2, vo.getNick());  
        pstmt.setString(3, vo.getEmail());  
        pstmt.setString(4, vo.getHp());  
        pstmt.setString(5, vo.getZip());  
        pstmt.setString(6, vo.getAddr1());  
        pstmt.setString(7, vo.getAddr2());  
        pstmt.setString(8, vo.getUid());  
  
        result = pstmt.executeUpdate();  
  
        close();  
    } catch (Exception e) {  
        logger.error(e.getMessage());  
    }  
    logger.debug("result : " + result);  
    return result;  
}
```

```
// 회원정보 수정  
public static final String UPDATE_USER_INFO = "UPDATE `board_user` SET "  
+ "    `name`=?, "  
+ "    `nick`=?, "  
+ "    `email`=?, "  
+ "    `hp`=?, "  
+ "    `zip`=?, "  
+ "    `addr1`=?, "  
+ "    `addr2`=? "  
+ " WHERE `uid`=?"
```

문제 4 – 회원정보 설정

3. 회원 수정하기 기능 구현 - 시연화면

name	nick	email	hp	grade	zip	addr1	addr2
서정현	허니	ooo33425@naver.com	010-3140-9982	2	61421	광주 동구 계림동 538-38	202호



회원정보 설정

아이디

sjh940520

비밀번호

비밀번호 입력

비밀번호 확인

비밀번호 입력 확인

비밀번호 수정

회원가입일

2022-10-21 16:50:31

개인정보 수정

이름

서정

별명

공백없는 한글, 영문, 숫자 입력

허니2

중복확인

이메일

ooo33425@naver.com

인증번호 받기

휴대폰

010-3140-9982

인증번호 받기

주소

61421

우편번호 찾기

광주 동구 계림동 538-38

202호

회원탈퇴

탈퇴하기



name	nick	email	hp	grade	zip	addr1	addr2
서정	허니2	ooo33425@naver.com	010-3140-9982	2	61421	광주 동구 계림동 538-38	203호

감사합니다