

# 포트폴리오

## 발표

Follow Me 인스타마켓 판매회사를 위한 그룹웨어

이다예

# 목차

프로젝트 개요

세부기능 소개

담당기능 시연

코드 리뷰

Q&A

# 프로젝트 개요

**Follow Me : 인스타마켓 판매회사를 위한 그룹웨어 개발 프로젝트**

## 기획의도

인스타마켓에서 패션 관련 상품을 판매하는 중소 기업을 가정해  
회사 규모 및 분위기에 적합한 그룹웨어 기능을 제공하고자 함

## 수행기간

2024.04.08 - 2024.06.07

기획 및 화면/DB 설계(3주), 화면 구현(1주)  
기능 구현 (3주), 테스트 (1주)

## 개발환경

운영체제 : Windows OS

개발환경 : Spring, Visual Studio Code

DBMS : Oracle DB - sqldeveloper

Server : Apache Tomcat 9.0

개발언어 : Java SE 11, HTML5, CSS, JavaScript, jQuery

라이브러리 및 기술 : JSP, JSTL, Bootstrap

2차 프로젝트 그룹웨어 메인화면

# 세부기능 소개

## 메인 페이지



로그인

마이페이지

쪽지함

메일

알림

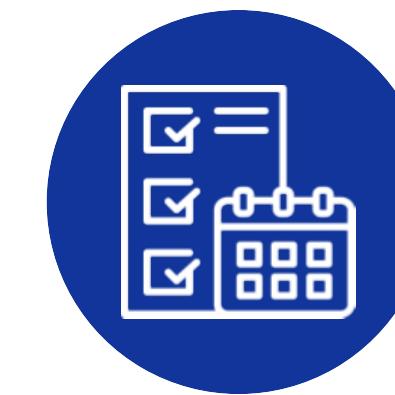


전자게시판

조직도

소셜피드

전자결재



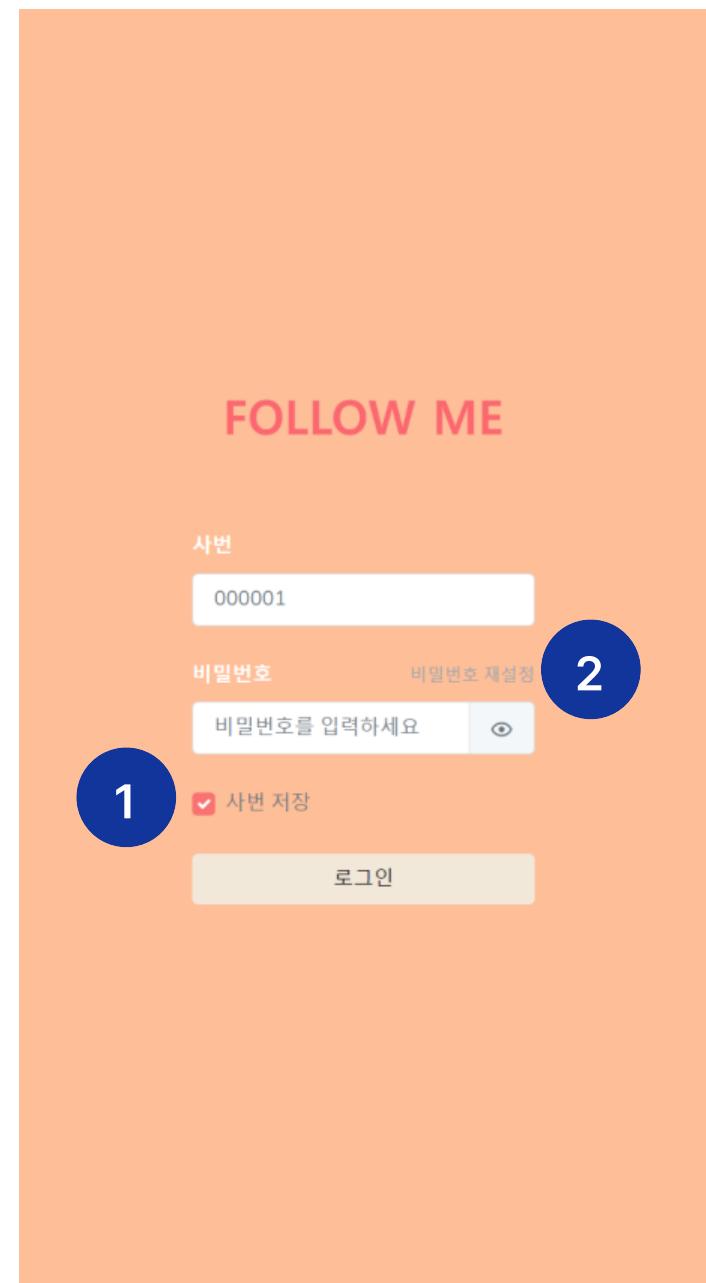
일정 관리

근태 관리

자산 관리

인사 관리

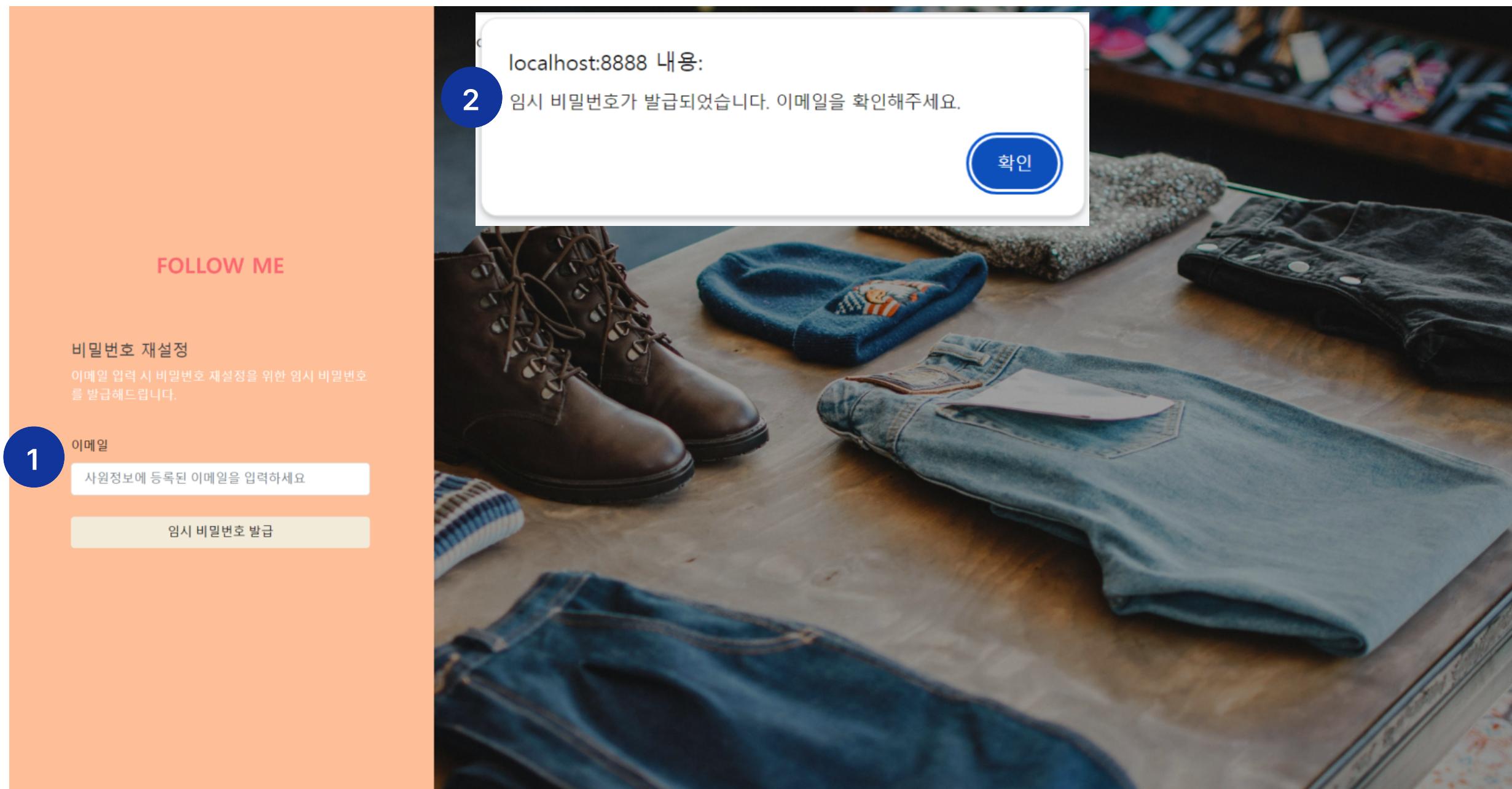
# 담당기능 시연 - 로그인



1 : Cookie를 활용해 사번 저장

2 : [비밀번호 재설정] 클릭 시  
임시비밀번호 발급 페이지로 이동

# 담당기능 시연 - 비밀번호 재설정



1 : 이메일 입력 시 DB에서 실제 존재하는 사용자인지 확인

2 : DB에 존재하는 회원일 경우 사용자가 입력한 이메일 주소로 임시 비밀번호 발송 처리  
(JavaMailSender 활용)

# 담당기능 시연 - 메인페이지

The screenshot displays a dashboard with the following components and numbered callouts:

- FOLLOW ME** sidebar (Left): Includes links for 조직도, 계시판, 일정관리, 자산관리, 전자결재, 인사관리, 이메일, 소셜피드, 근태관리, and 직원근태.
- Top Bar:** Includes a Google search bar, a notification bell, a search bar for employee names, and a profile icon for '이다에 님'.
- Module 1: 근태관리 (Left):** Shows a clock icon at 07:24:30 PM, a '출근 전' button, and two buttons for '출근하기' and '퇴근하기'.
- Module 2: 오늘의 결재 (Bottom Left):** Displays four buttons: 미처리 0건, 진행중 0건, 승인 0건, and 반려 0건.
- Module 3: 최근 공지글 (Center Left):** A table with columns '제목' and '작성일'. Items include:
 

제목	작성일
[공지]휴가 신청 및 유의사항	2024-05-26
[공지]긴급 서버점검 안내	2024-02-20
[공지]긴급 회의 소집 안내	2024-02-20
[공지]2024년 연차 신청 안내	2024-02-15
[공지]업무 프로젝트 완료 보고	2024-02-10
- Module 4: 오늘의 일정 (Bottom Center):** Includes a '새로운 일정 추가하기' button and a '추가' button.
- Module 5: 최근 판매왕 (Right):** Shows a chart for '하민수' with the title '금주의 판매왕' and a small profile picture.
- Module 6: 오늘의 패션 뉴스 (Bottom Right):** Displays news articles from Naver News API related to 패션. One article is highlighted in blue.

1 : 전자게시판의 최근 공지글을 조회  
& 각 게시글 링크 연결 처리

2 : 오늘의 To-do 리스트 등록/삭제

3 : DB의 주문정보/제품정보/직원  
테이블을 조인해 최근 1주일 내  
판매 실적이 가장 높은 직원 표시

4 : 네이버 뉴스 API를 활용  
제목에 '패션'이 들어간 뉴스만을  
표시하도록 Javascript 활용

5 : 구글 사이트번역 API 활용

# 담당기능 시연 - 마이페이지

The screenshot shows the 'MyPage' interface with various sections and features highlighted by numbered circles (1-5).

- 1 : 프로필 이미지 클릭 시 변경 가능**
- 2 : 전자서명 파일 업로드 시 변경 가능**
- 3 : 이메일, 비밀번호 등 개인정보 변경**
- 4 : 포트원 API를 활용해 은행명(은행 코드)과 계좌번호로 예금주를 조회한 후 예금주를 반환하도록 처리**
- 5 : 보유자격증 등록/수정/삭제 및 자격증 증빙 첨부파일 업로드 가능**

1 : 프로필 이미지 클릭 시 변경 가능

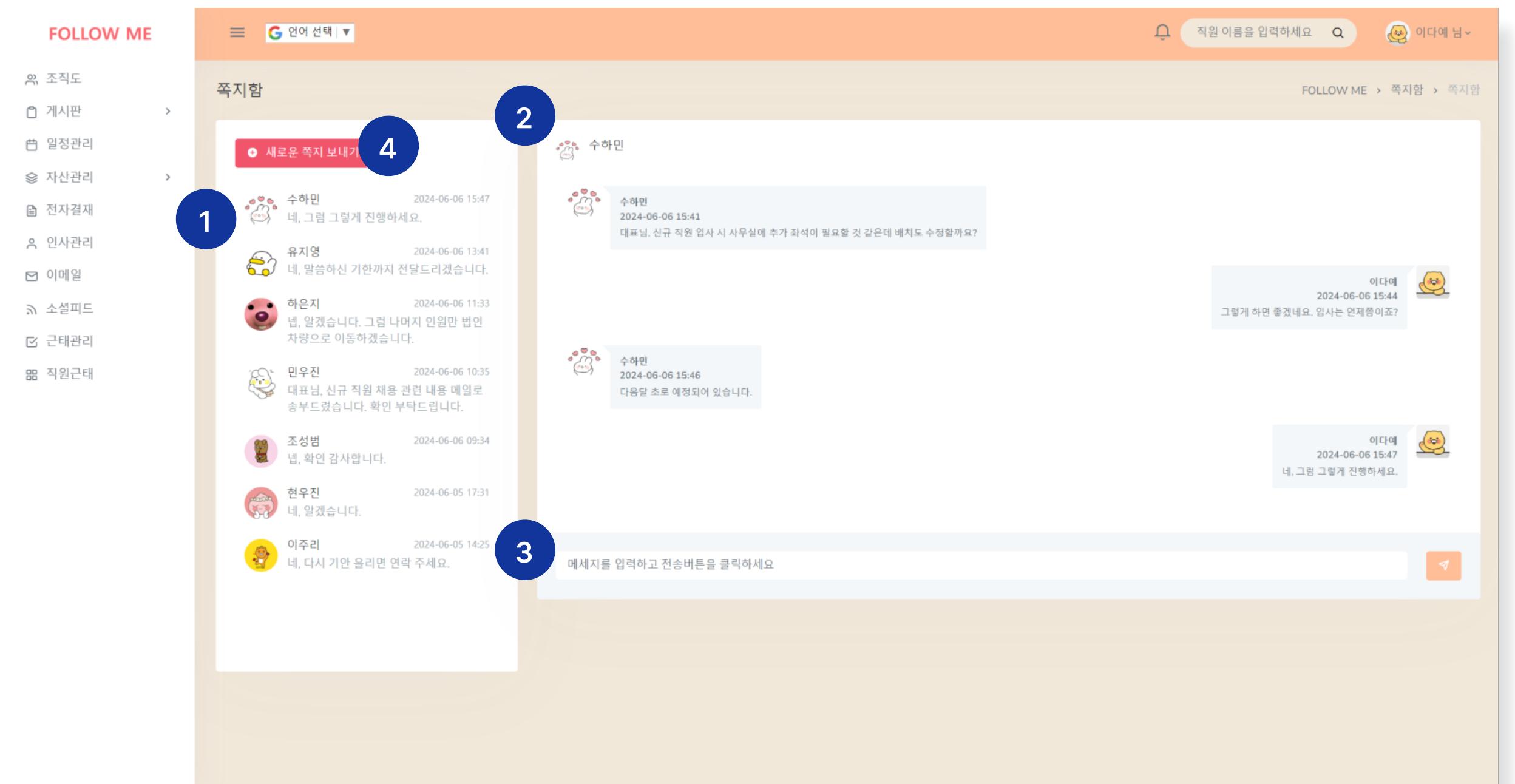
2 : 전자서명 파일 업로드 시 변경 가능

3 : 이메일, 비밀번호 등 개인정보 변경

4 : 포트원 API를 활용해 은행명(은행  
코드)과 계좌번호로 예금주를 조회  
한 후 예금주를 반환하도록 처리

5 : 보유자격증 등록/수정/삭제 및  
자격증 증빙 첨부파일 업로드 가능

# 담당기능 시연 - 쪽지함



1 : 쪽지함 클릭 시 Ajax로 이전 쪽지 리스트가 조회됨

2 : 개별 쪽지 리스트 클릭 시 Ajax로 이전 대화가 조회되며 표시됨

3 : 신규 메시지 전송 시 화면에 표시됨  
과 동시에 왼쪽 쪽지 리스트에도  
동기화되면서 쪽지함 리로드 됨

4 : [새로운 쪽지 보내기] 클릭 시 직원 목록이 팝업창으로 뜨며 선택 가능.  
로그인한 본인 및 기존에 존재하는  
쪽지함 리스트의 직원은 선택 불가.

# 담당기능 시연 - 인사관리

사원명	사원번호	권한레벨	부서	직급	직무	계약연봉	입사일	상태	수정/삭제
이다예	000001	3	영업팀	대표	사업 총괄	9,000만원	2020-02-23	<span>재직</span>	<span>수정</span> <span>삭제</span>
이주리	000002	1	영업팀	팀장	영업 총괄	7,000만원	2024-03-01	<span>재직</span>	<span>수정</span> <span>삭제</span>
현우진	000003	1	마케팅팀	팀장	마케팅 총괄	7,000만원	2020-04-30	<span>재직</span>	<span>수정</span> <span>삭제</span>
조성범	000004	2	경영지원팀	팀장	경영지원 총괄	6,800만원	2020-07-17	<span>재직</span>	<span>수정</span> <span>삭제</span>
은우진	000005	1	물류팀	팀장	물류 총괄	6,800만원	2020-09-11	<span>재직</span>	<span>수정</span> <span>삭제</span>
김민정	000006	1	대기발령	팀장	-	6,500만원	2020-10-02	<span>재직</span>	<span>수정</span> <span>삭제</span>
민우진	000008	0	마케팅팀	매니저	온라인 광고 담당	5,500만원	2020-12-19	<span>재직</span>	<span>수정</span> <span>삭제</span>

1 : [인사관리] 메뉴 클릭 시 Ajax로 직원 정보 조회해서 화면에 표시

2 : 검색 또한 비동기로 처리

3 : 신규 직원정보 등록 버튼을 통해 신규 정보 등록이 가능하며, 수정 및 삭제도 가능함

4 : 원하는 직원 체크박스 선택 후 엑셀로 내역 내려받기 가능함

# 코드 리뷰 - 사번 저장

## MemberController.java

```

} else if (bcryptPwdEncoder.matches(m.getMemPwd(), loginUser.getMemPwd())) {
    request.getSession().setAttribute("loginUser", loginUser);
    request.getSession().setAttribute("userAtt", userAtt);

    // 로그인 시 사번 저장 (쿠키 저장)
    if ("SAVE".equals(request.getParameter("memNoSaveCheck"))) {
        Cookie cookie = new Cookie("savedMemNo", loginUser.getMemNo());
        cookie.setMaxAge(30 * 24 * 60 * 60); // 30일 저장 유효기간
        cookie.setPath("/");
        response.addCookie(cookie);
    } else {
        Cookie cookie = new Cookie("savedMemNo", loginUser.getMemNo());
        cookie.setMaxAge(0);
        cookie.setPath("/");
        response.addCookie(cookie);
    }

    out.println("alert('" + loginUser.getMemName() + "님 환영합니다.');" );
    out.println("location.href = '" + request.getContextPath() + "/ceoMain.page';");
} else {
    out.println("alert('로그인에 실패했습니다. 사번 및 비밀번호를 다시 확인해주세요.');" );
    out.println("history.back();");
}

```

1 : 사용자가 사번을 저장하길 원하면 (memNoSaveCheck 파라미터가 "SAVE"라면) 사번을 30일 동안 저장하는 쿠키를 생성하고, 그렇지 않다면 쿠키를 삭제함 (MaxAge를 0으로 설정)

# 코드 리뷰 - 임시 비밀번호 발송

## MemberController.java

```
// 임시 비밀번호 발급 및 입력한 이메일로 발송
@PostMapping("/resetPwd.do")
public String resetPwd(String memEmail, RedirectAttributes redirectAttributes) {

    // 사용자가 입력한 이메일로 존재하는 사용자인지 확인
    MemberDto isMember = memberService.memEmailCheck(memEmail);

    if (isMember != null) {

        Random r = new Random();
        int tempPwd = r.nextInt(900000)+100000; // 랜덤 난수

        if (isMember.getMemEmail().equals(memEmail)) {
            EmailDto email = EmailDto.builder()
                .emailFrom("chennychat@gmail.com")
                .emailTo(memEmail)
                .subject("[Follow Me] 임시 비밀번호가 발급되었습니다.")
                .text(String.format("안녕하세요 %s님\n발급된 임시 비밀번호는 %d입니다.", isMember.getMemName(), tempPwd))
                .build();

            // 새로운 임시 비밀번호(암호화 필요)를 DB에 업데이트
            String hashedPassword = bcryptPwdEncoder.encode(String.valueOf(tempPwd)); // 비밀번호 해싱
            isMember.setHashedPassword(hashedPassword);

            memberService.updateTempPwd(isMember);

            // 임시 비밀번호 발급 이메일 발송
            emailSender.sendMail(email);
            redirectAttributes.addFlashAttribute("alertMsg", "임시 비밀번호가 발급되었습니다. 이메일을 확인해주세요.");
            return "redirect:/";
        }
    }
}
```

1 : 사용자가 입력한 이메일로 DB에  
존재하는 사용자인지 확인

DB에 존재하는 직원이 맞다면,  
랜덤 6자리 숫자를 생성함

추가 확인으로 사용자의 이메일이  
입력한 이메일과 일치하는지 확인

이메일 전송을 위한 EmailDto  
객체를 빌더 패턴을 사용해 생성

임시 비밀번호를 해싱 후 해싱된  
비밀번호를 사용자 객체에 설정  
하고 DB에 비밀번호를 업데이트

# 코드 리뷰 - 메인페이지 뉴스 API

## ChartController.java

```

@GetMapping("/naverNews.do")
public ResponseEntity<String> getNaverNews() {
    String clientId = " "; // 애플리케이션 클라이언트 아이디
    String clientSecret = " "; // 애플리케이션 클라이언트 시크릿
    String query = "패션"; // 검색어

    String encodedQuery = java.net.URLEncoder.encode(query, StandardCharsets.UTF_8);
    String apiUrl = "https://openapi.naver.com/v1/search/news.json?query=" + encodedQuery + "&display=100&start=1&sort=date";

    HttpHeaders headers = new HttpHeaders();
    headers.add("X-Naver-Client-Id", clientId);
    headers.add("X-Naver-Client-Secret", clientSecret);

    RequestEntity<Void> req = RequestEntity
        .get(URI.create(apiUrl))
        .headers(headers)
        .build();

    RestTemplate restTemplate = new RestTemplate();
    ResponseEntity<String> response = restTemplate.exchange(req, String.class);
    return response;
}

```

1 : 네이버 API를 사용하기 위한 애플리케이션 클라이언트 아이디/시크릿 키 및 검색어로 "패션"을 설정함

네이버 뉴스 검색 API의 URL을 설정함 - 검색어, 표시할 결과 수, 시작 위치, 정렬 방식 등을 포함

RestTemplate 객체를 생성해 네이버 API에 요청을 보내고 응답을 String 형태로 받음

# 코드 리뷰 - 마이페이지 계좌실명인증

## MemberController.java

```
// 마이페이지 - 계좌실명인증 기능
@PostMapping("/checkAccount")
public ResponseEntity<Map<String, String>> checkAccount(@RequestBody Map<String, String> requestData) {
    String bankCode = requestData.get("bank_code");
    String bankNumber = requestData.get("bank_num");

    Map<String, String> response = new HashMap<>();
    try {
        Map<String, String> map = checkAccountService.getAccessToken1(bankCode, bankNumber);
        String bankHolderInfo = map.get("bankHolderInfo");
        Object errorObj = map.get("error");
        if (errorObj instanceof String) {
            String errorStr = (String) errorObj;
            int error = Integer.parseInt(errorStr);
            response.put("errormsg", String.valueOf(error));
        }
        response.put("bankHolderInfo", bankHolderInfo);
        return ResponseEntity.ok(response);
    } catch (Exception e) {
        response.put("errormsg", e.getMessage());
        return ResponseEntity.status(HttpStatus.INTERNAL_SERVER_ERROR).body(response);
    }
}
```

1 : 요청데이터에서 은행코드와 계좌 번호를 추출하고, 응답 데이터를 저장할 Map 객체를 생성

checkAccountService를 호출하여 계좌 인증을 수행.  
해당 메서드는 은행 코드와 계좌 번호를 받아 해당 계좌의 실명정보를 반환함

오류 발생 시 적절한 오류 메시지를 반환함

# Q&A