

Pet Integration Platform Development



“반려동물의 삶을 이롭게, 당신의 삶을 행복하게.”

목차

1. 기획 의도

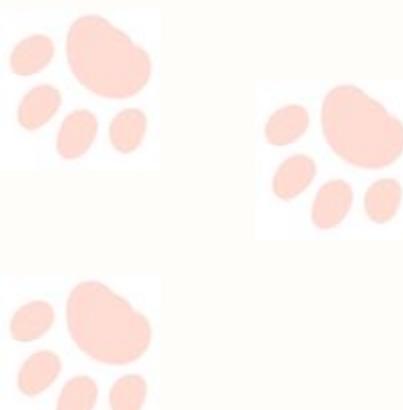
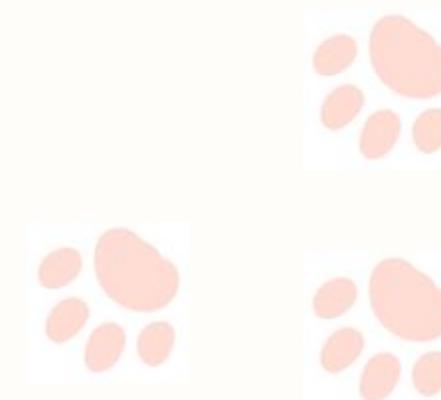
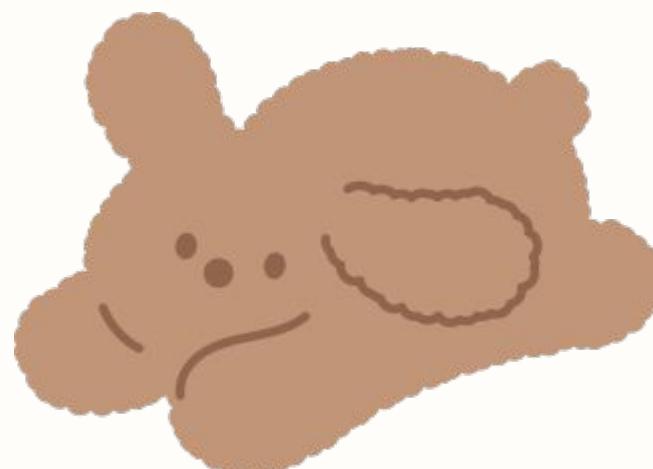
2. 프로젝트 개요

3. 화면 설계

4. ERD

5. 기능 구현

6. 시연



01

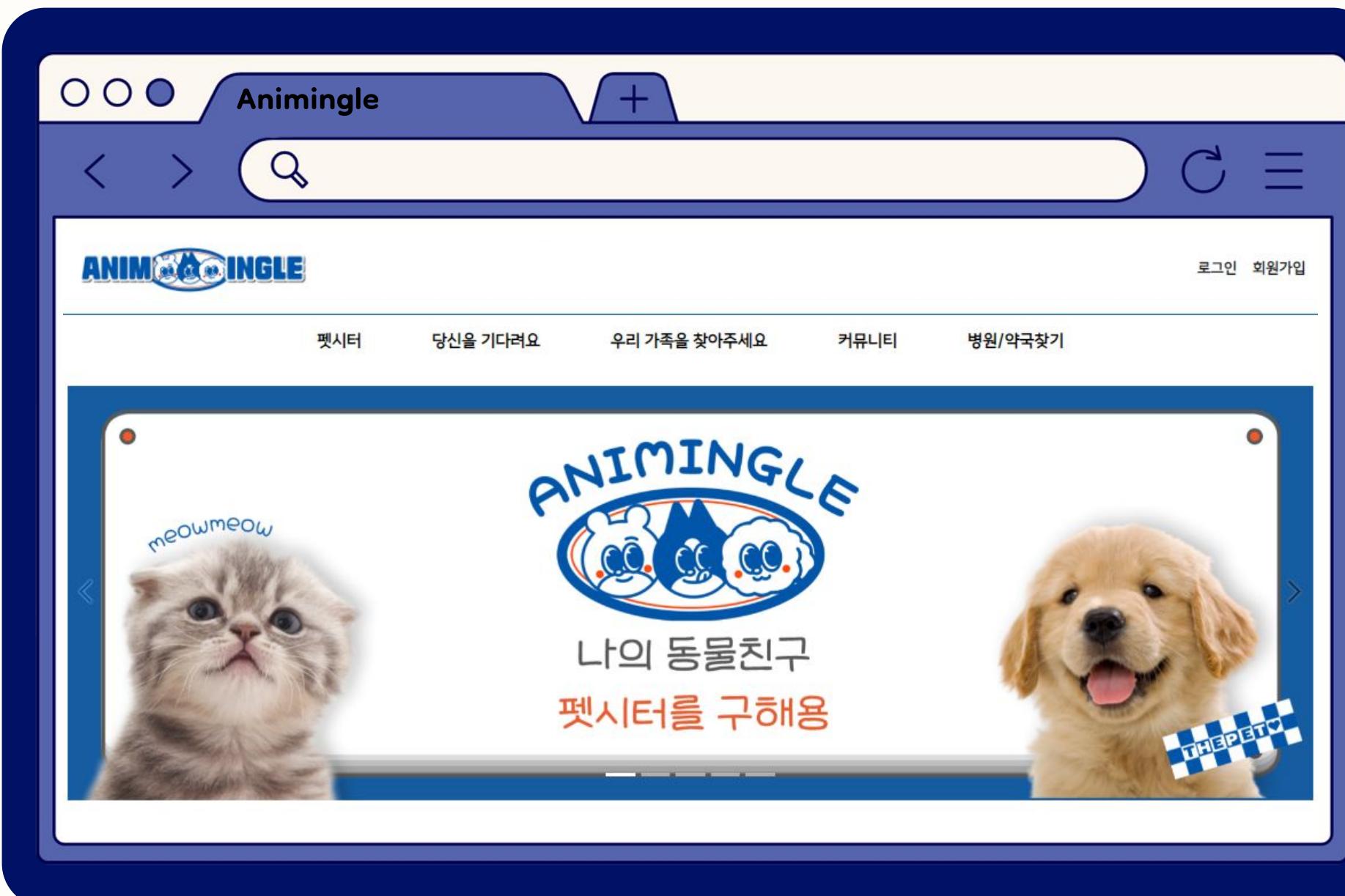
기획 의도

웹페이지 제작의도

반려동물 키우는 인구수의 증가로 인해 반려동물 관련 서비스에 대한 수요가 높아지고 있다.

하지만 반려동물의 다양한 정보성과, 다양한 기능을 보유하고 있는 웹사이트들은 현재 존재하지 않았고, 비슷한 현행 플랫폼에서는, 그 플랫폼들의 요구사항에 맞춘 기능들의 부재들이 다수 존재하였다는 의견이 많았다.

이러한 문제점들을 통해, 우리 "애니밍글"은 다양한 반려동물 관련 기능을 한 플랫폼에 통합하여 반려동물을 사랑하는 사람들에게 요구사항에 맞춘 더욱 편리한 서비스와 정보를 제공하고자 제작하게 되었다.



선행 연구 관련 플랫폼 및 비교



포인핸드

전국 동물 보호소의 유기동물과
사람을 이어주는 플랫폼

동물보호 관리시스템에서 데이터를
받아와 유기동물 공고들을 확인하고,
해당 공고마다 댓글을 달 수 있는
기능을 제공해주고 있는 플랫폼이다.

이외에 개인이 분실, 목격, 구조한
동물들에 대한 글을 따로 작성할 수
있고, 구조동물들의 분양글도 작성이
가능하다.

하지만, 기능의 다양성이 미흡한
점으로 인해 다소 아쉬움이 크다.

너랑

반려동물 콘텐츠 플랫폼

각종 커뮤니티(일상, 재미, 질문 등),
산책로 데이트, 카페 데이트, 지역별
데이트, 동물 병원, 지식, 쇼핑, 훈련
정보 등 다양한 서비스를 제공해주고
있는 플랫폼이다.

하지만, 현재 서비스가 원활히
제공되지 않고 있고,
부분부분 미흡한 부분과 에러가 있어
다소 아쉬움이 크다.

비마이펫

반려동물 콘텐츠 플랫폼

지식정보 콘텐츠, 반려동물 용품
쇼핑, 커뮤니티 서비스 등을
제공하고 있으며 사용자가 직접
제작한 콘텐츠(웹툰, 영상 등)를
공유할 수 있다는 것이 특징이다.

그러나 익명 커뮤니티 특성 상
신뢰할 만한 충분한 정보를
제공받기가 힘들고, 사용자들이
무분별한 정보를 받을 수 있다는
점이 다소 아쉬움이 크다.

펫 플래닛

가정집 펫시터 예약 서비스

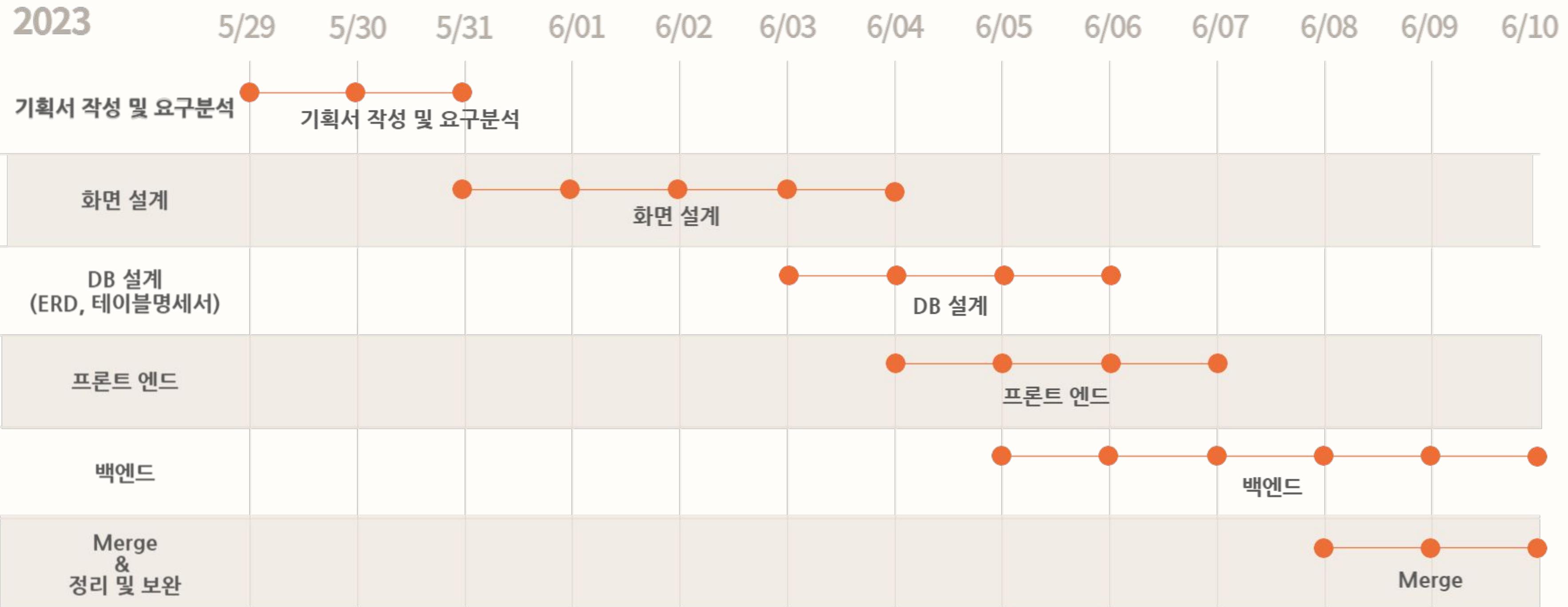
인증이 완료된 펫시터들이 자신의
집이나 장소에서 애완 동물을
맡아주는 서비스로, 이용하고자 하는
회원이 원하는 일자, 장소들을
선택하면 일정 금액에 따라 결제 및
매칭을 해주는 플랫폼이다.

하지만, 금액적인 부분에서 다소
높은 금액이 측정이 되어있고, UI,
UX인 부분에서 아쉬운 점, 기능의
다양성이 미흡한 점으로 다소
아쉬움이 크다.

02

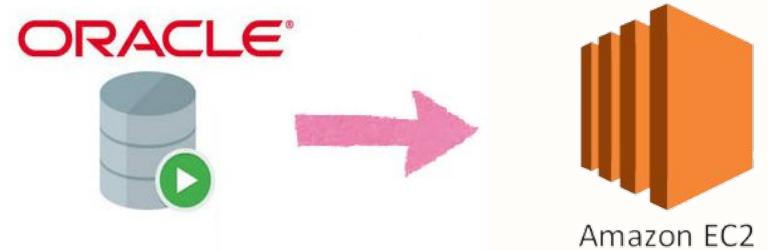
프로젝트 개요

프로젝트 일정



프로젝트 사용 기술

데이터베이스



AWS 데이터베이스

화면 설계



피그마

협업 툴



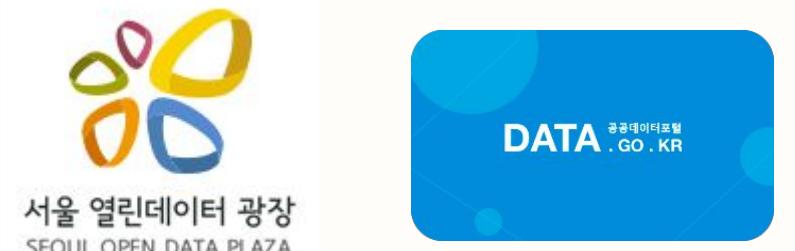
Git, SourceTree, Github

ERD



exERD

API



서울 열린데이터 광장
SEOUL OPEN DATA PLAZA

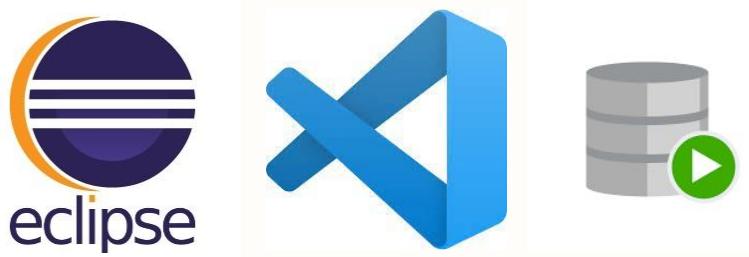
서울 열린 데이터 광장
&
공공데이터 포털

WAS



Apache Tomcat

개발 툴



EClipse
Visual Studio Code
Sql Developer



HTML, CSS, JS, AJAX,
JAVA(Servlet)

03

화면 설계

화면설계 메인화면/회원가입/로그인



메인화면

The registration page has the ANIMINGLE logo at the top left. Navigation links are at the top right. The main content is a '회원가입' (Registration) form. It includes fields for '아이디(6~12자, alphabet, number)', '비밀번호(alphabet, number 6~20, Case)', '비밀번호 확인', '닉네임(2~10글자)', '이 름(한글 2~8자)', '생년 월 일 <input type="calendar">', '010 — 4853 — 7923', 'email @ gmail.com', '주소(API request)', '상세주소', '반려동물' (with '유 / 무' radio buttons, '유' is checked), '사 진' (with file input 'pic.png'), and a 'Create Account' button. A small note at the bottom says 'Already have an account? Log in'.

회원가입

The login page has the ANIMINGLE logo at the top left. Navigation links are at the top right. The main content is a '로그인' (Login) form. It includes fields for '아이디' and '비밀번호'. Below the fields are social media login buttons for KakaoTalk (yellow), Naver (green), and Google (blue). A large blue button labeled '로그인' is at the bottom.

로그인

화면설계 마이페이지(공통)

ANIMINGLE

로그아웃 마이페이지 고객센터

회원 정보

회원 정보
펫시터 모집 내역
인증 센터
내 문의사항

이름: 장기성
닉네임: 2지존기성z
주소: 서울시 강남구 역삼동
한국별당 8층
연락처: 010-1234-5678
사진수정
수정

현재 비밀번호
새 비밀번호 일련
새 비밀번호 일련
새 비밀번호 확인 일련
비밀번호 변경하기

내 게시글: 100건
내 댓글: 30건

ABOUT PET LIFE

회원 정보

ANIMINGLE

로그아웃 마이페이지 고객센터

펫시터 모집 내역

회원 정보
펫시터 모집 내역
인증 센터
내 문의사항

내 펫시터 모집 내역

골든 리트리버 3살 3일간 맡아주실 분~	모집중 2023-05-12
천사같은 고양이.. 맡아주실 분...	완료 2023-05-10
밥만 쟁겨주실 분 구해요ㅠㅠ	완료 2023-05-12
악마 고양이 봐주실 분..	완료 2023-05-12
우리집 강아지 진짜 착해요 맡아주세요	완료 2023-04-01

ABOUT PET LIFE

펫시터 모집내역

ANIMINGLE

로그아웃 마이페이지 고객센터

인증 센터

회원 정보
펫시터 모집 내역
인증 센터
내 문의사항

동물등록 칩	인증하기
펫시터	인증완료(2023-05-23)
수의사	인증하기

ABOUT PET LIFE

인증 센터

ANIMINGLE

로그아웃 마이페이지 고객센터

내 문의사항

내 문의사항
회원 정보
펫시터 모집 내역
인증 센터
내 문의사항

[신고] 이상한 사진올리는 놈이 있어요	확인중 2023-05-12
[문의] 펫시터 인증이 잘 안되네요..	답변 완료 2023-05-10
[신고] 못생겼어요...	답변 완료 2023-05-12
[문의] 프로필 사진 올리는데 에러 나요	답변 완료 2023-05-12
[신고] 사기꾼이 있는 것 같아요	답변 완료 2023-04-01

→ 해당 유저 탈퇴 처리하겠습니다. 불편을 드려 죄송합니다.

ABOUT PET LIFE

내 문의사항

화면설계 마이페이지(인증 회원 프로필)

The screenshot shows the 'Certification Center' section of the Animalingle My Page. It includes a sidebar with icons for 회원 정보 (Member Information), 펫시터 모집 내역 (Pet Sitter Job History), 인증 센터 (Certification Center), and 내 문의사항 (My Inquiry). The main area features a large input field for '동물 등록 번호 입력' (Enter Animal Registration Number) with placeholder '12345678910' and a '등록' (Register) button. A small circular icon with a paw print is visible in the bottom right corner.

인증 화면

The screenshot shows the 'Pet Sitter Profile' section of the Animalingle My Page. It includes a sidebar with icons for 회원 정보 (Member Information), 펫시터 프로필 (Pet Sitter Profile), 펫시터 모집 내역 (Pet Sitter Job History), 인증 센터 (Certification Center), and 내 문의사항 (My Inquiry). The main area displays the profile of a pet sitter named 이민지 (Lee Minji) with a 5.0 rating and 999 reviews. It includes a photo, a self-introduction message, and a list of job applications.

펫시터 회원 프로필

The screenshot shows the 'Veterinarian Profile' section of the Animalingle My Page. It includes a sidebar with icons for 회원 정보 (Member Information), 수의사 프로필 (Veterinarian Profile), 펫시터 모집 내역 (Pet Sitter Job History), 인증 센터 (Certification Center), and 내 문의사항 (My Inquiry). The main area displays the profile of a veterinarian named 장기성 (Jang Gi Sung) with a photo, contact information, and a list of messages.

수의사 회원 프로필

화면설계 게시판 목록

The screenshot shows the 'Petter' (펫시터) board section of the ANIMOMINGLE website. At the top, there are navigation links: 로그아웃, 마이페이지, 고객센터, 펫시터, 당신을 기다려요, 우리 가족을 찾아주세요, 커뮤니티, 병원/약국찾기. Below this is a search bar with the placeholder '나의 동물친구 펫시터를 구해요'. Underneath the search bar are three buttons: 모집중 (Job Post), 모집 완료 (Job Completed), and 전체보기 (View All). A '지역' (Region) dropdown menu is also present. The main content area displays four job posts from users: 1. 링글맹글(mingle77) 16분 전: 골든 리트리버 3살 3일간 맡아주실 분~ [2023.05.03~2023.05.05] 서울시 강남구 역삼동. 2. 애니멀러버(animal486) 30분 전: 천사같은 고양이 맡아주실 분 구해요~~~~~ [2023.05.03 08:00~22:00] 서울시 강남구 역삼동. 3. 애니멀러버(animal486) 30분 전: 악마같은 고양이 맡아주실 분 구해요 [2023.05.30~2023.05.31] 서울시 강남구 역삼동. 4. 애니멀러버(animal486) 1일 전: 밥만 쟁겨주실 분 구해요ㅠㅠ [2023.05.03 08:00~22:00] 서울시 강남구 역삼동. Each post includes a '모집종료' (Job Ended) button. At the bottom right is a '작성하기' (Post) button with a pencil icon, and at the bottom left are page navigation buttons: Previous, 1, 2, 3, 4, 5, Next.

펫시터 게시판

The screenshot shows the 'Woogwoog' (와글와글) board section of the ANIMOMINGLE website. At the top, there are navigation links: 로그아웃, 마이페이지, 고객센터, 펫시터, 당신을 기다려요, 우리 가족을 찾아주세요, 커뮤니티 (highlighted in red), 병원/약국찾기. Below this is a search bar with the placeholder '와글와글'. Underneath the search bar are four buttons: 일상 (Daily), 정보 공유 (Information Share), 나눔 (Sharing), and 전체보기 (View All). A '지역' (Region) dropdown menu is also present. The main content area displays five posts from users: 1. 링글맹글(mingle77) 16분 전: [일상] 우리 얘기 돌잔치 했어용 [54 replies, 1 comment]. 2. 애니멀러버(animal486) 30분 전: [정보공유] 여기 식물 키우시는 분은 없나요? [54 replies, 1 comment]. 3. 애니멀러버(animal486) 30분 전: [나눔] 스크래쳐 나눔합니당(역삼동) [54 replies, 1 comment]. 4. 애니멀러버(animal486) 30분 전: [일상] 우리 막내 자랑합니다^^ [54 replies, 1 comment]. Each post includes a '모집종료' (Job Ended) button. At the bottom right is a '작성하기' (Post) button with a pencil icon, and at the bottom left are page navigation buttons: Previous, 1, 2, 3, 4, 5, Next.

와글와글 게시판

The screenshot shows the 'Vet Q&A' (수의사 Q&A) board section of the ANIMOMINGLE website. At the top, there are navigation links: 로그아웃, 마이페이지, 고객센터, 펫시터, 당신을 기다려요, 우리 가족을 찾아주세요, 커뮤니티 (highlighted in red), 병원/약국찾기. Below this is a search bar with the placeholder '수의사 Q&A'. Underneath the search bar are four buttons: 강아지 (Dog), 고양이 (Cat), 기타 (Others), and 전체보기 (View All). A '지역' (Region) dropdown menu is also present. The main content area displays five posts from users: 1. 강아지가 먹으면 안돼는 음식 [답변: 0, 7시간 전] 2. 고양이 딸기 먹어도 되나요? [답변: 2, 22시간 전] 3. 강아지 석류 먹여도 돼나요? [답변: 2, 1일 전] 4. 길고양이 키워도 되나요?? [답변: 1, 1일 전] Each post includes a '답변' (Reply) button. At the bottom right is a '작성하기' (Post) button with a pencil icon, and at the bottom left are page navigation buttons: Previous, 1, 2, 3, 4, 5, ..., 6288, Next.

수의사 Q&A 게시판

화면설계 상세보기 목록

ANIMINGLE

로그아웃 마이페이지 고객센터

펫시터 당신을 기다려요 우리 가족을 찾아주세요 커뮤니티 병원/약국찾기

밍글랭글(mingle77)

[골든 리트리버 3살 3일간 맡아주실 분~] 16분 전

서울시 강남구 역삼동 2023.05.03 ~ 2023.05.05

3살 된 골든 리트리버입니다. 3개월 아닙니다.
순하고 착해요. 산책만 하루에 2번 시켜 주시고 사료는 소분해서 잘 보이는 곳에 뒀습니다.
산책 꼭 시켜주셔야 됩니다ㅠㅠ 아이가 밖에서만 배변을 해서...
부탁 드리겠습니다.
사례비: 15만원
olo-48s3-tq23 연락 부탁드립니다ㅠ

수정하기 삭제하기

신청자 목록

펫시터1(ilovedog99)
돌봄 횟수: 15회
별점: 4.5

강아지 사랑
나라 사랑
나라에서 허락한 유일한 마약 = 강아지

ABOUT PET LIFE

펫시터 상세보기

ANIMINGLE

로그아웃 마이페이지 고객센터

펫시터 당신을 기다려요 우리 가족을 찾아주세요 커뮤니티 병원/약국찾기

밍글랭글(mingle77)

[일상] 우리 애기 돌잔치 했어용 16분 전

귀엽종~~
개껌 집었어요.
돌상은 제가 직접 차려짚답니다😊

수정하기 삭제하기

2개의 댓글

하루하루(daily33) 10분 전

애기 넘 귀여워용ㅠㅠ 저희 애기 돌잔치 때 생각나네요~~~~~

B I M S E T I C O P Q D R -

생각의 차이를 인정하고 공감을 나눠주세요.

목록 보기 댓글 작성

ABOUT PET LIFE

와글와글 상세보기

ANIMINGLE

로그아웃 마이페이지 고객센터

펫시터 당신을 기다려요 우리 가족을 찾아주세요 커뮤니티 병원/약국찾기

밍글랭글(mingle77)

[강아지] 낯선 사람이 오면 짖어요 16분 전

문 밖에 사람 인기척만 나도 하루종일 짖어요...
너무 짖어서 목이 다 쉬었어요... 하지 말라고 간식을 줘도 오히려 더 짖는 것 같네요ㅠㅠ

수정하기 삭제하기

답변

10분 전

청각이 특히 예민한 아이가 있습니다.
사람 인기척에 익숙해 질 수 있도록 꾸준한 훈련을 해주면 좋을 것 같습니다.
인기척이 날 때 긍정적 보상을 주는 방법이 있어요~~~
심각해지면 저희 병원에 데려오세요

수의사1(docter77)
참조아동들병원...
답변 횟수: 90회

개인정보를 공유 및 요청하거나, 명예훼손, 무단 광고, 불법 정보 유포시 모니터링 후 삭제될 수 있으며,
이에 대한 민형사상 책임은 게시자에게 있습니다.

목록 보기 댓글 작성

ABOUT PET LIFE

수의사 Q&A 상세보기

화면설계 게시판 목록

ANIMINGLE

로그아웃 마이페이지 고객센터

펫시터 당신을 기다려요 우리 가족을 찾아주세요 커뮤니티 병원/약국찾기

산책 친구 구하기

Map showing dog walking spots (Pet Spots) in a neighborhood. A green line highlights a specific route or area.

ABOUT PET

산책친구 구해요 게시판

ANIMINGLE

로그아웃 마이페이지 고객센터

펫시터 당신을 기다려요 우리 가족을 찾아주세요 커뮤니티 병원/약국찾기

온누리약국 영업중 강남구 역삼동~~

Map showing the location of Onuri Pharmacy (온누리약국) in Gwangjin-gu, Seoul, near the Yaksan Station area.

ABOUT PET LIFE

병원/약국 찾기 게시판

화면설계 상세보기 목록

ANIMOGGLE

로그아웃 마이페이지 고객센터

[펫시터](#) [당신을 기다려요](#) [우리 가족을 찾아주세요](#) [커뮤니티](#) [병원/약국찾기](#)

산책 친구 구하기

ABOUT PET

산책친구 구해요 상세보기

병원/약국 찾기 상세보기

화면설계 게시판 목록

The screenshot shows a grid of 15 small animal posts, each featuring a cartoon animal head icon and the text "Animingle". Below each icon is a list of four items: 이름(센터), 줄, 품종, and 성별. At the bottom of the page, there are navigation links for "Previous", "1 2 3 4 5", and "Next".

당신을 기다려요 게시판

The screenshot shows a list of lost pet posts. The first post is from "밍글 맹글(mingle77)" about a lost dog found in Gangnam-gu, Yeksam-dong. The second and third posts are from "애니멀러버(animal486)" about a lost dog found in Nonhyeon Station, Line 6. The fourth post is also from "애니멀러버(animal486)" about the same lost dog. At the bottom right, there is a "작성하기" (Write) button.

우리 가족을 찾아주세요 게시판

화면설계 상세보기 목록

The screenshot shows a lost pet report for a dog named "당신을 기다려요". The report includes the following details:

1. 동물의 정보

종별번호	경남-진주-2022-04294
종	개
종종	믹스견
나이	2세04개월
성별	암컷
체중	2.0kg
임시보호내용	호스미스 임시보호 가능합니다.
소개영상URL	https://youtu.be/L5_BrIc64Vu

2. 동물보호센터 관내

관할보호센터명	경주시유기동물보호소
주소	경상남도 경주시 갑현면 산당길 207번길 22 (갑현면, 자작동집개장사점)
전화번호	055-749-6194
관할기관	경상남도 경주시
담당자	경주시청
책임자	055-748-6204

목록 보기

ABOUT PET LIFE

당신을 기다려요 상세보기

The screenshot shows a found pet report for a dog named "우리 가족을 찾아주세요". The report includes the following details:

[펫 찾아요] 강남구 역삼동에서 아이를 잃어버렸어요...

역삼역 3번 출구 쪽 역삼 개나리공원에서 산책하다가 강아지를 잃어버렸습니다.
회색 푸들이고 나이는 3살입니다.
3kg 정도 되는 작은 아이구요.. 낮을 가려서 사람을 보면 짖을 거예요.
이름은 모찌입니다.
빨간색 목걸이를 착고 있습니다.
찾으신 분들은 010-10134-0678로 연락부탁드립니다.

지도

2개의 댓글

ABOUT PET LIFE

우리 가족을 찾아주세요 상세보기

화면설계 글쓰기 목록

The screenshot shows the ANIMINGLE website's post creation interface. At the top, there are navigation links: 로그아웃, 마이페이지, 고객센터. Below that, there are links for 펫시터, 당신을 기다려요, 우리 가족을 찾아주세요, 커뮤니티, 병원/약국찾기. The main section is titled "와글와글". It includes fields for "말머리" (Category) set to "일상", "제목" (Title) with a placeholder "제목을 입력해주세요.", and "본문" (Body) with a rich text editor. A preview of the post content is shown below the editor. At the bottom right are "취소" and "등록" buttons.

ABOUT PET LIFE

일반 게시판 글쓰기

The screenshot shows the ANIMINGLE website's post creation interface for finding a walking buddy. At the top, there are navigation links: 로그아웃, 마이페이지, 고객센터. Below that, there are links for 펫시터, 당신을 기다려요, 우리 가족을 찾아주세요, 커뮤니티, 병원/약국찾기. The main section is titled "산책 친구 구하기". It includes a "제목" (Title) field with a placeholder "제목을 입력해주세요.", a map showing a walking route from "제주첨단과학기술단지" to "제주첨단과학기술단지", and two "본문" (Body) sections with rich text editors. A preview of the post content is shown below the editors. At the bottom right are "취소" and "등록" buttons.

ABOUT PET LIFE

산책 친구 구해요 글쓰기

The screenshot shows the ANIMGGLE website's post creation interface for finding a family member. At the top, there are navigation links: 로그아웃, 마이페이지, 고객센터. Below that, there are links for 펫시터, 당신을 기다려요, 우리 가족을 찾아주세요, 커뮤니티, 병원/약국찾기. The main section is titled "우리 가족을 찾아주세요". It includes a "말머리" (Category) dropdown set to "펫 찾아요", a "제목" (Title) field with a placeholder "제목을 입력해주세요.", a map of a city area with a red location marker, and two "본문" (Body) sections with rich text editors. A preview of the post content is shown below the editors. At the bottom right are "취소" and "등록" buttons.

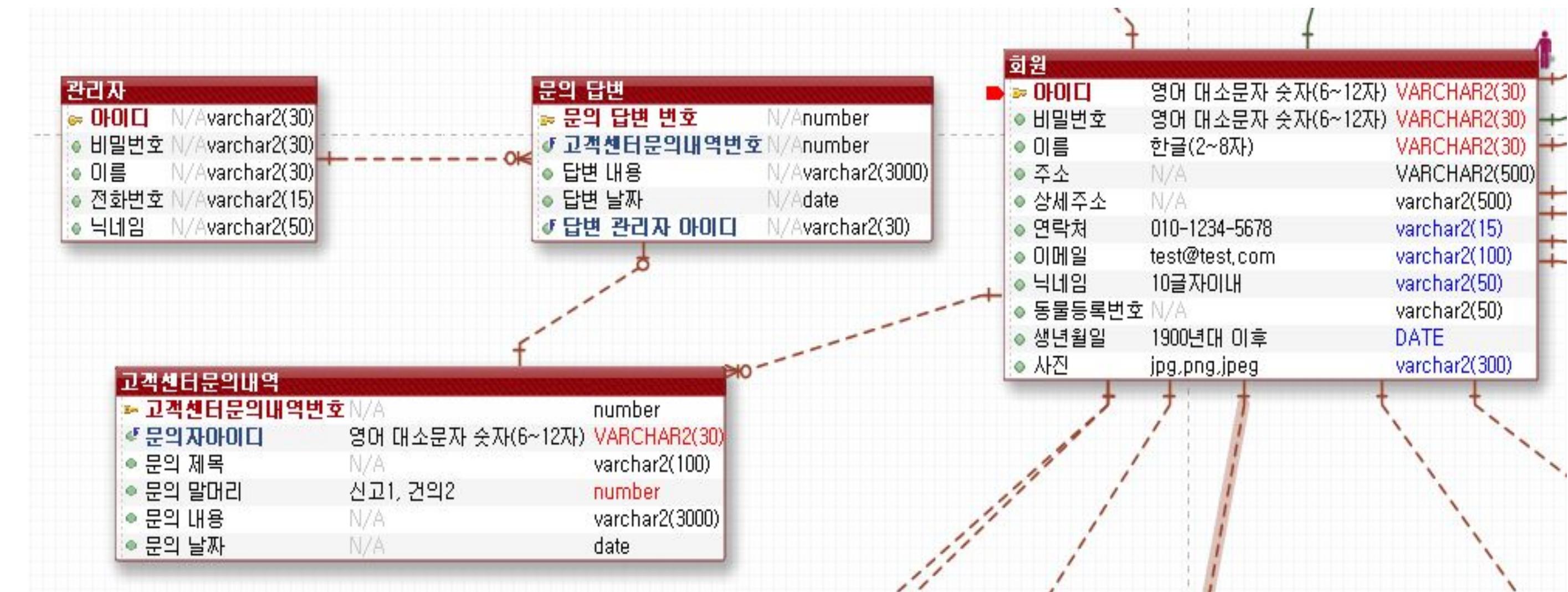
ABOUT PET LIFE

우리 가족을 찾아주세요 글쓰기

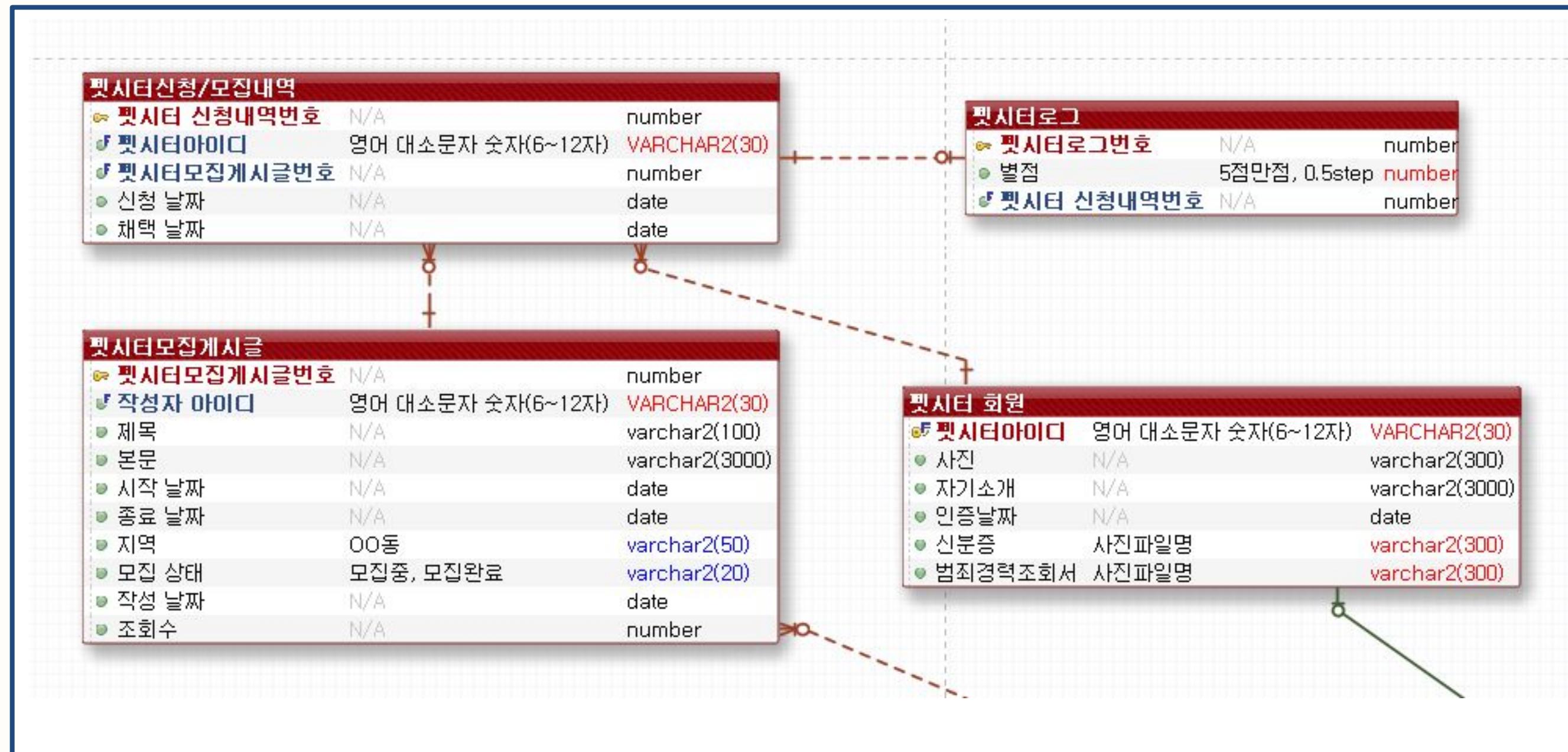
04

ERD

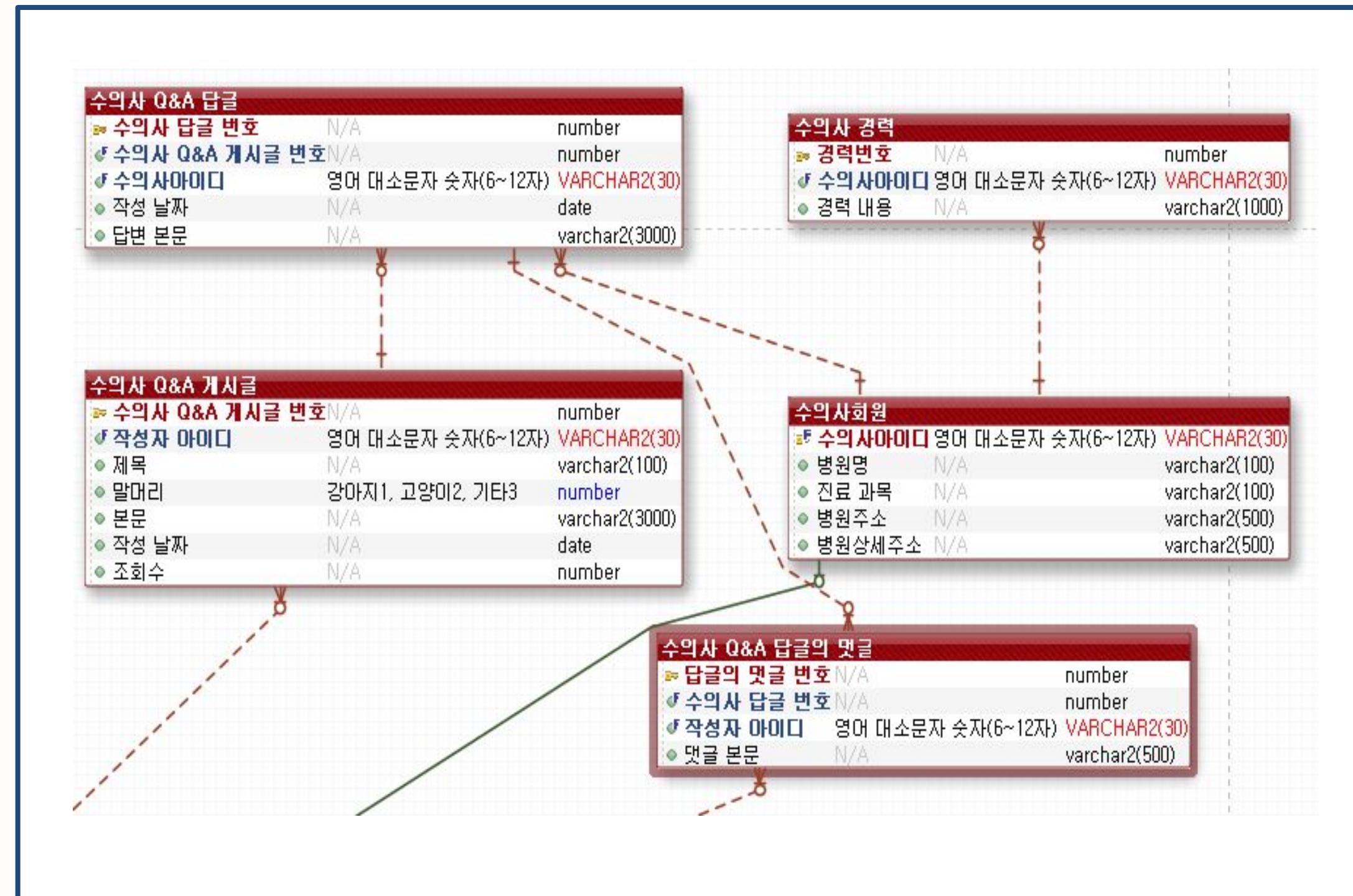
ERD - 회원, 관리자, 고객센터 문의내역



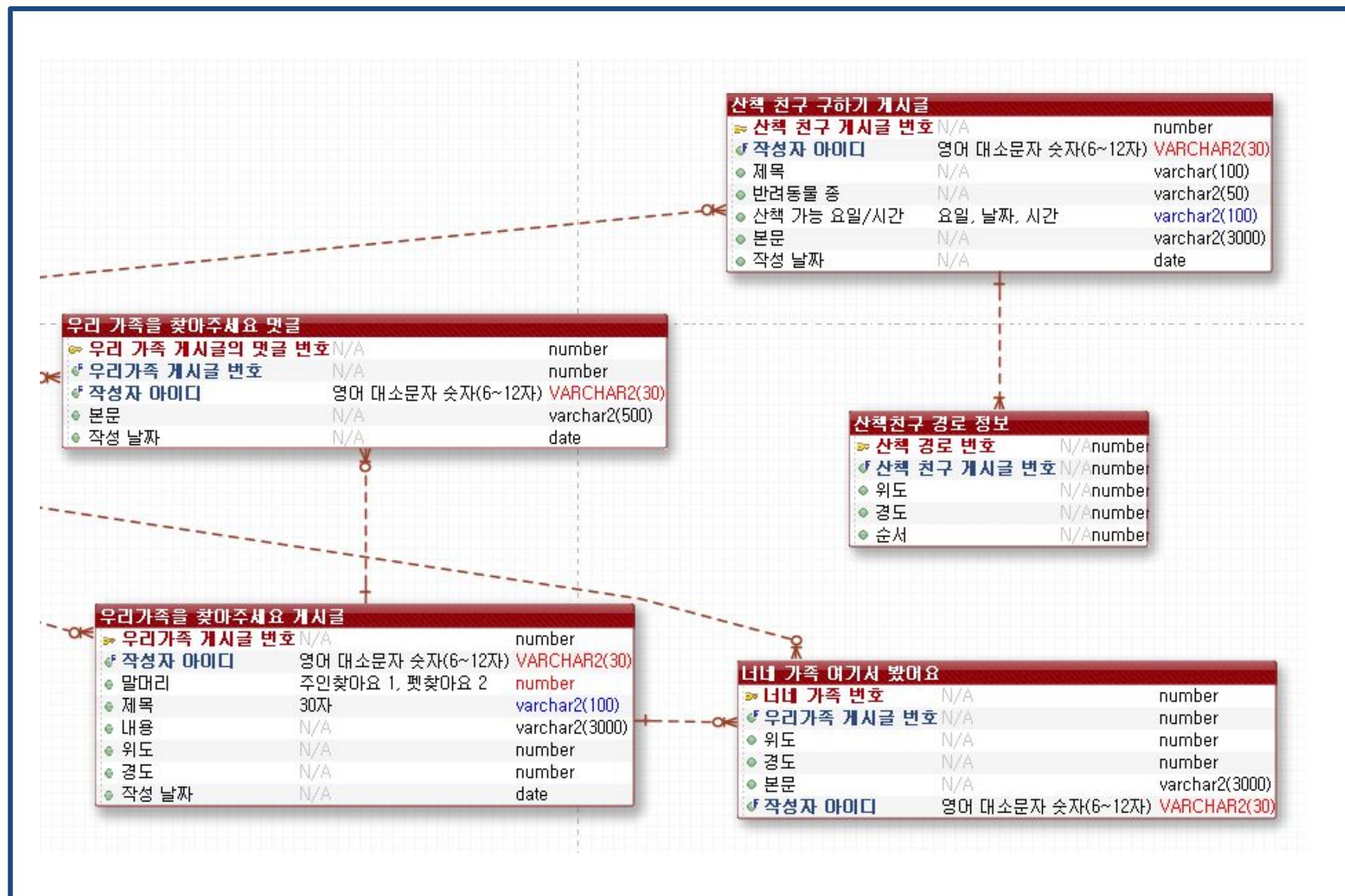
ERD - 팻시터



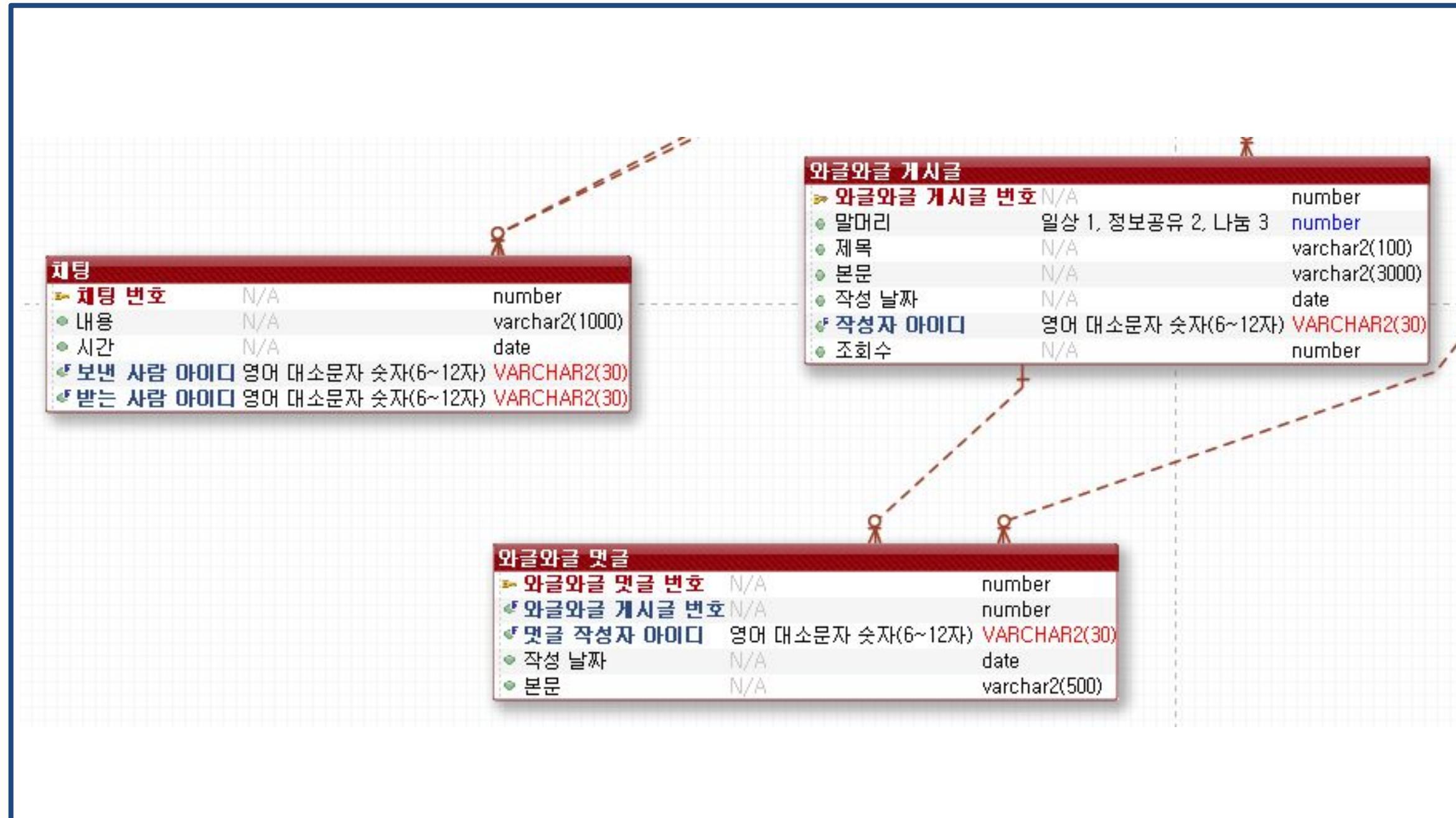
ERD - 수의사



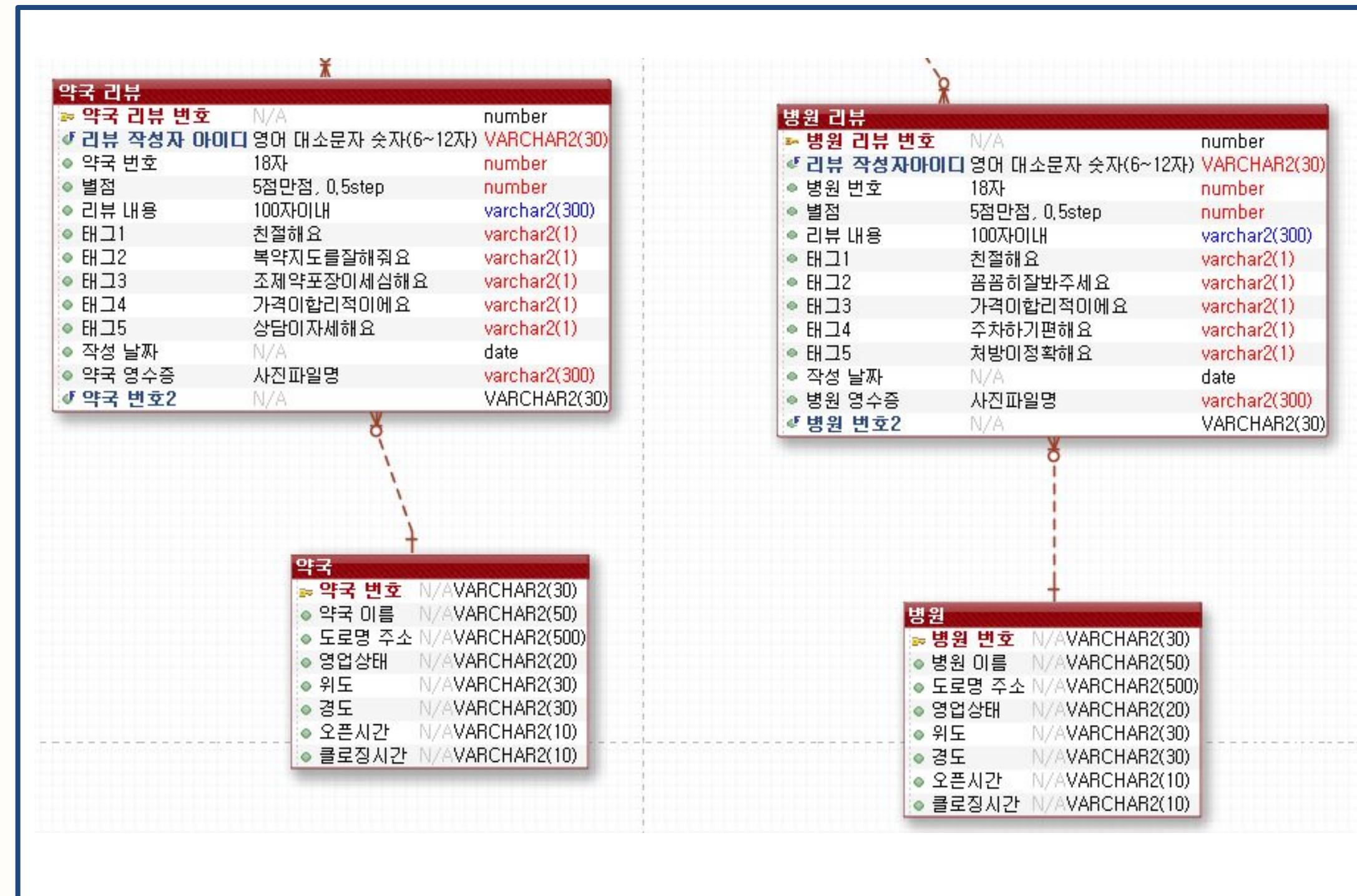
ERD - 산책친구, 우리가족 찾아주세요



ERD - 와글와글



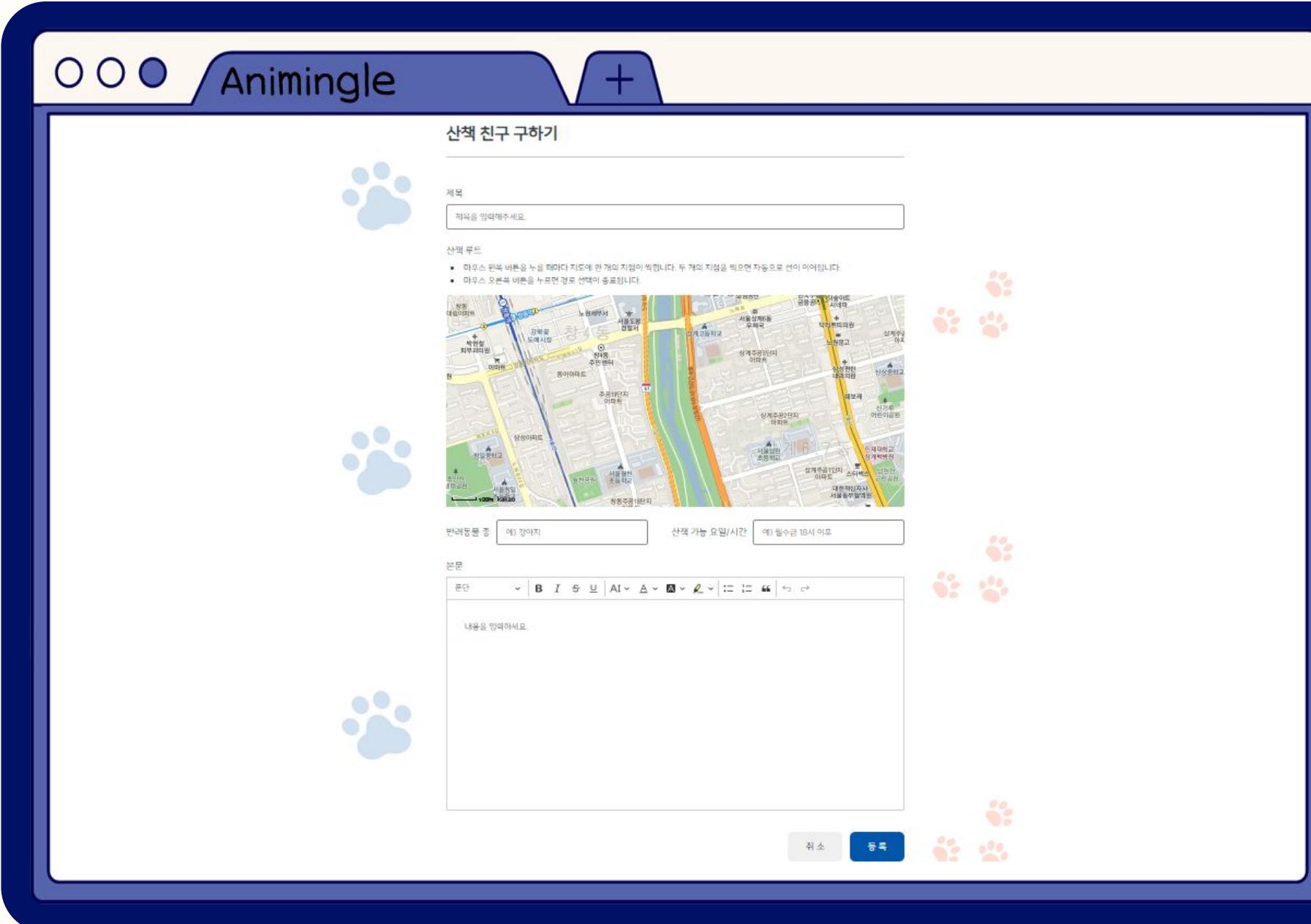
ERD - 약국, 병원, 리뷰



05

기능 구현

산책 친구 구하기



```

@Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    HttpSession session = req.getSession();

    String id = (String)session.getAttribute("id");
    String subject = req.getParameter("subject");
    String petKind = req.getParameter("petKind");
    String walkTime = req.getParameter("walkTime");
    String text = req.getParameter("text");
    String edit_seq = req.getParameter("seq");

    WalkTogetherDTO dto1 = new WalkTogetherDTO();
    dto1.setId(id);
    dto1.setWt_subject(subject);
    dto1.setWt_petkind(petKind);
    dto1.setWt_time(walkTime);
    dto1.setWt_content(text);

    WalkTogetherDAO dao = new WalkTogetherDAO();
    String[] pathLat = req.getParameterValues("pathLat");
    String[] pathLng = req.getParameterValues("pathLng");

    boolean isExist = dao.checkWriting(edit_seq);

    if(!isExist) {
        dao.addContent(dto1);

        String new_seq = String.valueOf(dao.getWtSeq());
        dto1.setWt_seq(new_seq);

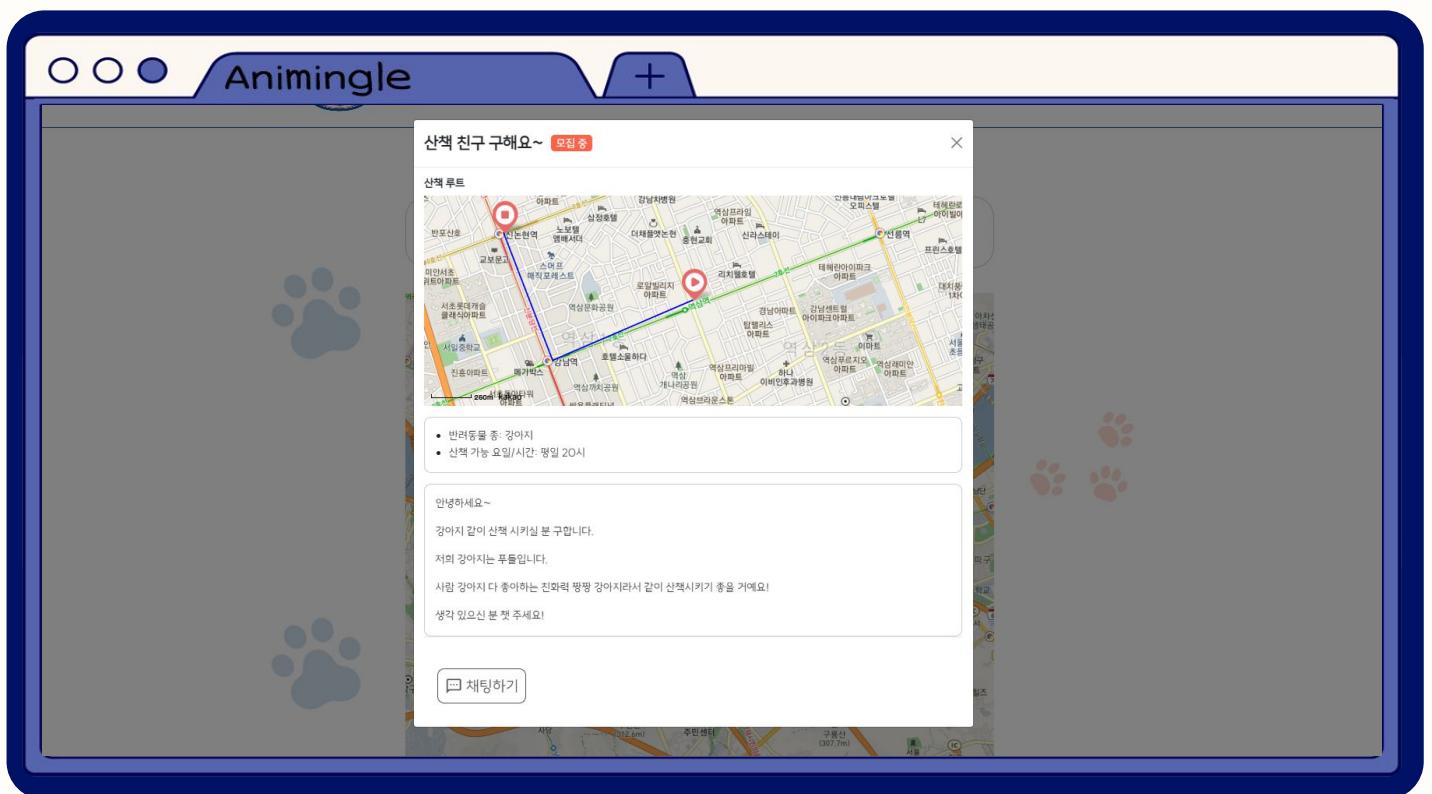
        dao.addPath(addPathDTO(pathLat, pathLng, new_seq));
    } else {
        dto1.setWt_seq(edit_seq);
        dao.editContent(dto1);

        if(pathLat != null && pathLng != null) {
            dao.delPath(edit_seq);
            dao.editPath(addPathDTO(pathLat, pathLng, edit_seq));
        }
    }

    resp.sendRedirect("/animingle/board/walktogetherlist.do");
}

```

산책 친구 구하기



```
$ajax({
  type: 'POST',
  url: '/animingle/board/walktogetherlist.do',
  data: {
    swLat: sw.getLatitude(),
    swLng: sw.getLongitude(),
    neLat: ne.getLatitude(),
    neLng: ne.getLongitude()
  },
  dataType: 'json',
  success: (result)=>{
    var positions = [];

    $(result).each((index, item)=>{
      var filteredPlist = item.plist.filter(subList => subList[0] === item.wt_seq);

      var filteredBeginPath = item.plist.filter(subList => subList[0] === item.wt_seq && subList[1] == '1');
      var beginLat = filteredBeginPath.map(subList => subList[2]);
      var beginLng = filteredBeginPath.map(subList => subList[3]);

      var filteredEndPath = item.plist.filter(subList => subList[0] === item.wt_seq);
      var maxOrder = Math.max(...filteredEndPath.map(subList => parseInt(subList[1])));
      var lastPath = filteredEndPath.filter(subList => subList[1] === maxOrder.toString());
      var endLat = lastPath.map(subList => subList[2]);
      var endLng = lastPath.map(subList => subList[3]);

      positions.push({
        petkind: item.wt_petkind,
        time: item.wt_time,
        beginlatlng: new kakao.maps.LatLng(beginLat, beginLng),
        endlatlng: new kakao.maps.LatLng(endLat, endLng),
        subject: item.wt_subject,
        content: item.wt_content,
        seq: item.wt_seq,
        pathlist: filteredPlist,
        writerid: item.wt_writer,
        sessionid: item.session_id
      });
    });
    showMarkernInfo(positions, bounds);
  },
  error: (a, b, c) => console.log(a, b, c)
});
```

당신을 기다려요



```
urlBuilder.append("/") + URLEncoder.encode("인증키", "UTF-8")); /* 인증키 (sample사용시에는
urlBuilder.append("/") + URLEncoder.encode("json", "UTF-8")); /* 요청파일타입 (xml, xmlf, xl
urlBuilder.append("/") + URLEncoder.encode("CardSubwayStatsNew", "UTF-8")); /* 서비스명 (i
urlBuilder.append("/") + URLEncoder.encode("1", "UTF-8")); /* 요청시작위치 (sample인증키 사
urlBuilder.append("/") + URLEncoder.encode("10", "UTF-8")); /* 요청종료위치 (sample인증키 사
// 상위 5개는 필수적으로 순서바꾸지 않고 호출해야 합니다.

// 서비스별 추가 요청 인자이며 자세한 내용은 각 서비스별 '요청인자' 부분에 자세히 나와 있습니다.
urlBuilder.append("/") + URLEncoder.encode("20220301", "UTF-8")); /* 서비스별 추가 요청인자

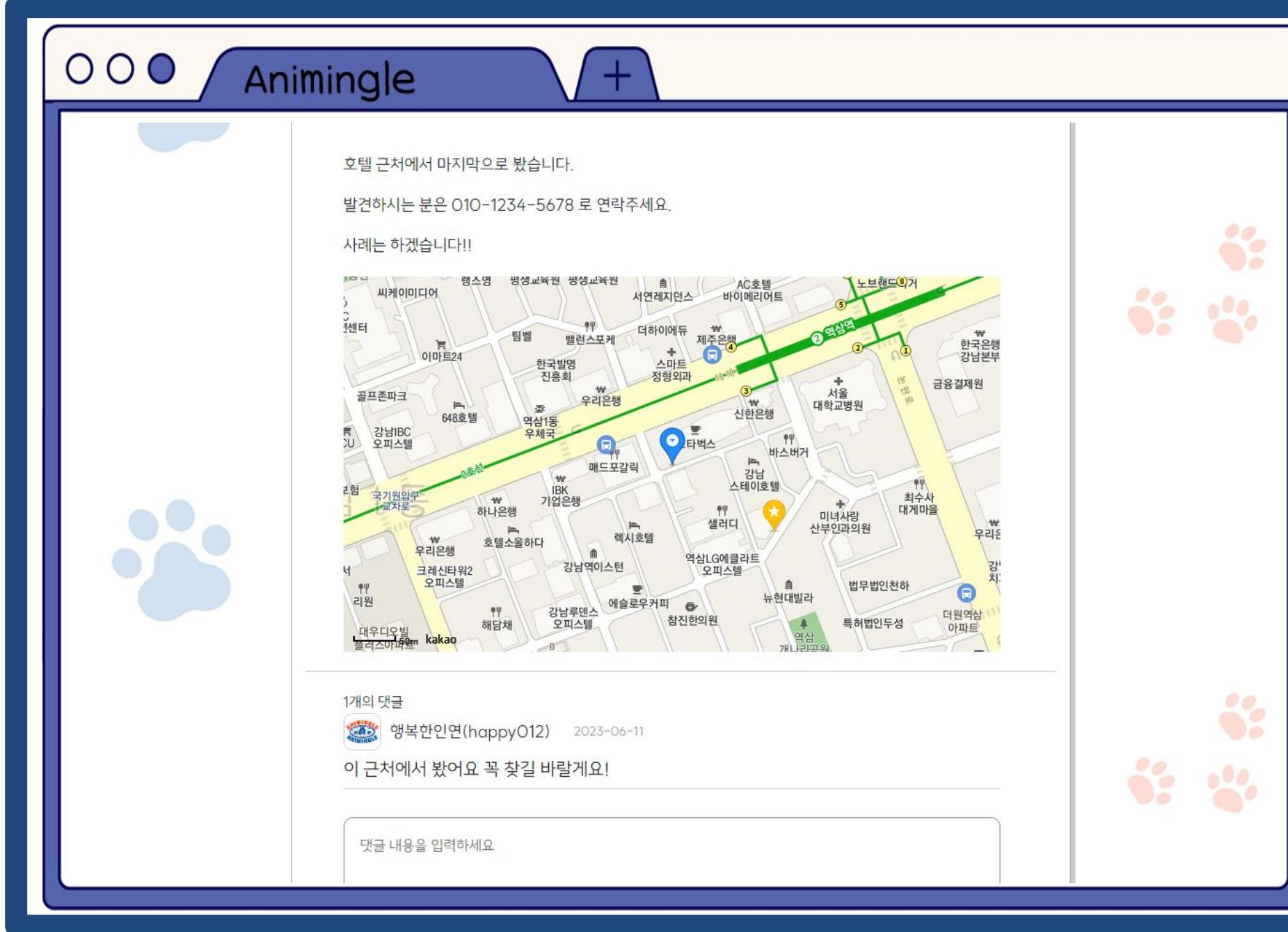
URL url = new URL(urlBuilder.toString());
HttpURLConnection conn = (HttpURLConnection) url.openConnection();
conn.setRequestMethod("GET");
conn.setRequestProperty("Content-type", "application/json");
//System.out.println("Response code: " + conn.getResponseCode()); /* 연결 자체에 대한 확인
BufferedReader rd;

// 서비스코드가 정상이면 200~300사이의 숫자가 나옵니다.
if (conn.getResponseCode() >= 200 && conn.getResponseCode() <= 300) {
    rd = new BufferedReader(new InputStreamReader(conn.getInputStream()));
} else {
    rd = new BufferedReader(new InputStreamReader(conn.getErrorStream()));
}
StringBuilder sb = new StringBuilder();
String line;
while ((line = rd.readLine()) != null) {
    sb.append(line);
}
rd.close();
conn.disconnect();

JSONParser parser = new JSONParser();
// 2. 문자열을 JSON 형태로 JSONObject 객체에 저장.
JSONObject obj = (JSONObject) parser.parse(sb.toString());
JSONObject jsonMain = (JSONObject) obj.get("TbAdpWaitAnimalView");

// 3. 필요한 리스트 데이터 부분만 가져와 JSONArray로 저장.
JSONArray dataArr = (JSONArray) jsonMain.get("row");
```

우리 가족을 찾아주세요



```
if (${fn:length(cplist)} != 0) {
    for (var i=0; i<${fn:length(cplist)}; i++) {

        var imageSize = new kakao.maps.Size(24, 35);

        var markerImage = new kakao.maps.MarkerImage(imageSrc, imageSize);

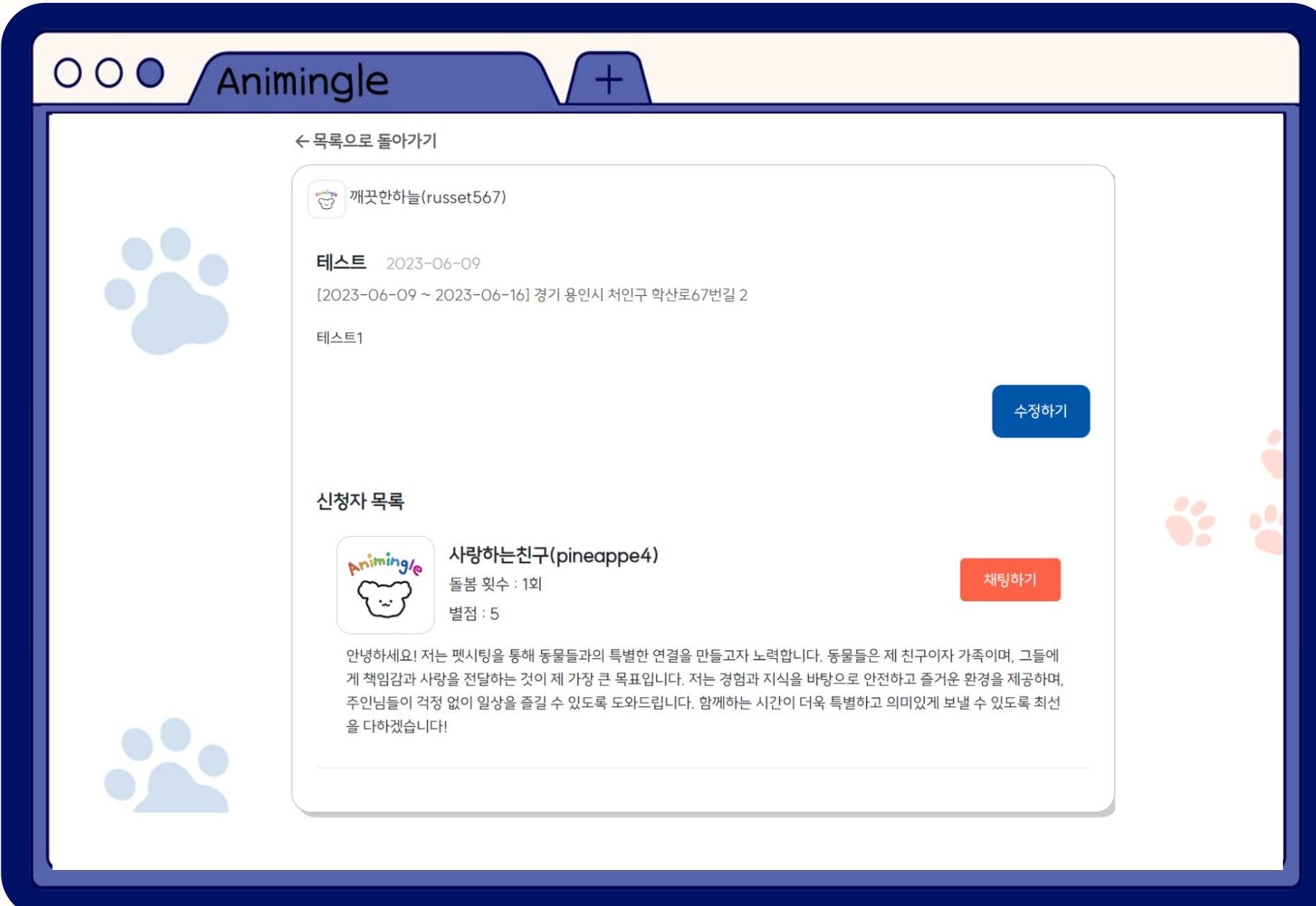
        var marker = new kakao.maps.Marker({
            map: map, // 마커를 표시할 지도
            position: positions[i].latlng, // 마커의 위치
            image : markerImage
        });

        // 마커에 표시할 인포윈도우를 생성합니다
        var infowindow = new kakao.maps.InfoWindow({
            content: positions[i].content // 인포윈도우에 표시할 내용
        });

        // 마커에 이벤트를 등록하는 함수 만들고 즉시 호출하여 클로저를 만듭니다
        // 클로저를 만들어 주지 않으면 마지막 마커에만 이벤트가 등록됩니다
        (function(marker, infowindow) {
            // 마커에 mouseover 이벤트를 등록하고 마우스 오버 시 인포윈도우를 표시합니다
            kakao.maps.event.addListener(marker, 'mouseover', function() {
                infowindow.open(map, marker);
            });

            // 마커에 mouseout 이벤트를 등록하고 마우스 아웃 시 인포윈도우를 닫습니다
            kakao.maps.event.addListener(marker, 'mouseout', function() {
                infowindow.close();
            });
       })(marker, infowindow);
    }
}
```

펫시터



```
@WebServlet("/board/petsitterapply.do")
public class PetsitterApply extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException,
    IOException {

        HttpSession session = req.getSession();
        String id = (String) session.getAttribute("id");
        String isPet = (String) session.getAttribute("isPet");
        String psr_seq = req.getParameter("psr_seq");
        String page = req.getParameter("page");

        PetsitterApplyDTO dto = new PetsitterApplyDTO();

        dto.setPs_id(id);
        dto.setPsr_seq(psr_seq);

        PetsitterDAO dao = new PetsitterDAO();
        List<PetsitterApplyDTO> dlist = dao.checkDuplication(psr_seq);

        resp.setCharacterEncoding("UTF-8");
        resp.setContentType("text/html");

        PrintWriter writer = resp.getWriter();

        for (PetsitterApplyDTO d : dlist) {
            if (d.getPs_id().equals(id)) {
                writer.print("<script>alert('이미 신청기록이 있습니다.');//</script>");
                writer.print("<script>location.href='/animingle/board/petsitterview.do?psr_seq=" +
                           psr_seq + "&page=" + page + "</script>");

                writer.close();
                return;
            }
        }

        if (id != null && isPet.equals("Y")) {
            dao = new PetsitterDAO();
            int result = dao.petApply(dto);

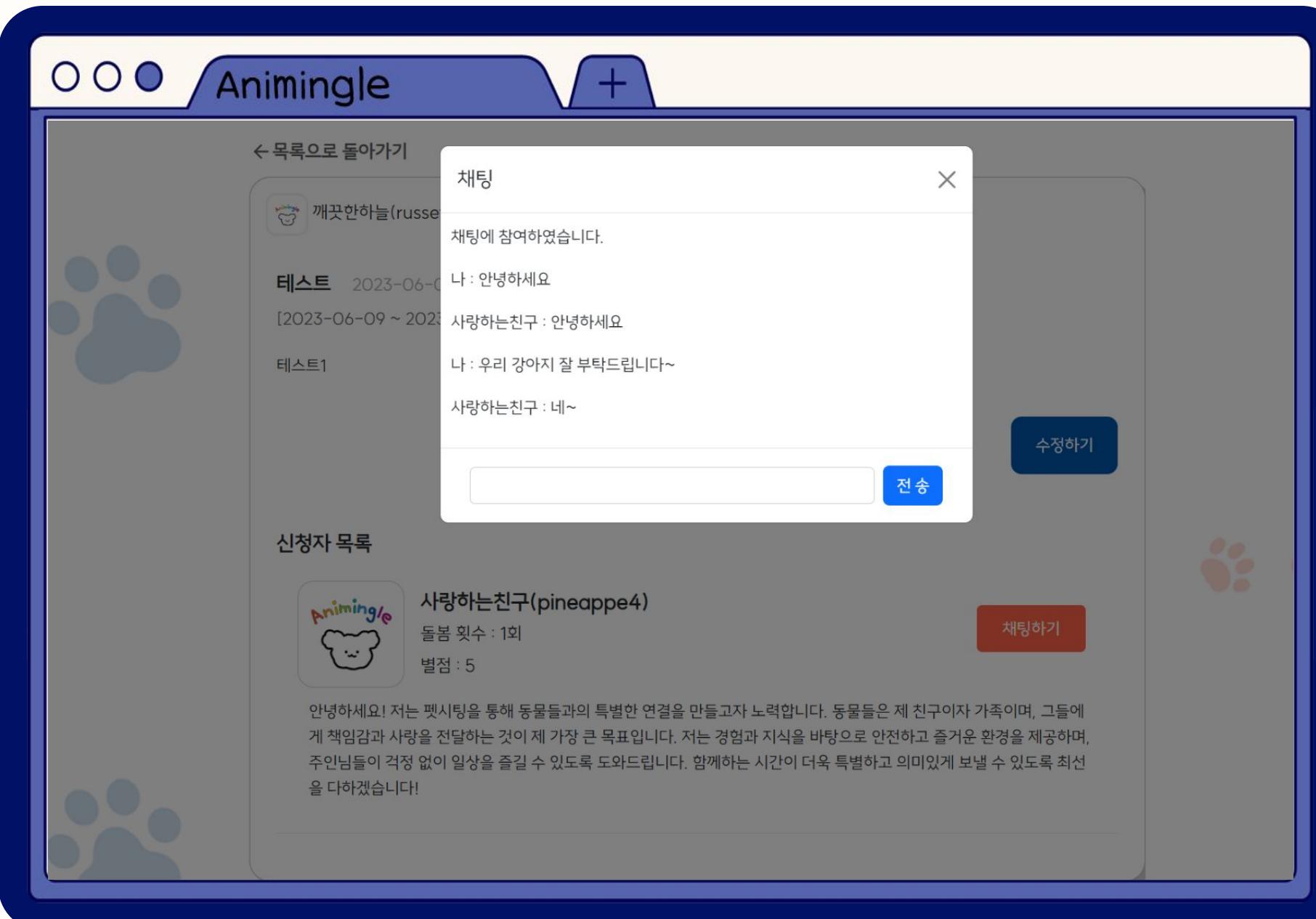
            if (result == 1) {
                writer.print("<script>alert('신청을 완료했습니다.');//</script>");
                writer.print("<script>location.href='/animingle/board/petsitterview.do?psr_seq=" +
                           psr_seq + "&page=" + page + "</script>");

                writer.close();
            } else {
                writer.print("<script>alert('신청을 실패했습니다.');//</script>");
                writer.print("<script>location.href='/animingle/board/petsitterview.do?psr_seq=" +
                           psr_seq + "&page=" + page + "</script>");

                writer.close();
            }
        } else {
            writer.print("<script>alert('펫시터 계정만 신청이 가능합니다.');//</script>");
            writer.print("<script>location.href='/animingle/board/petsitterview.do?psr_seq=" +
                           psr_seq + "&page=" + page + "</script>");

            writer.close();
        }
    }
}
```

펫시터



```
@ServerEndpoint("/petsitterview.do")
public class PetsitterChat extends HttpServlet {

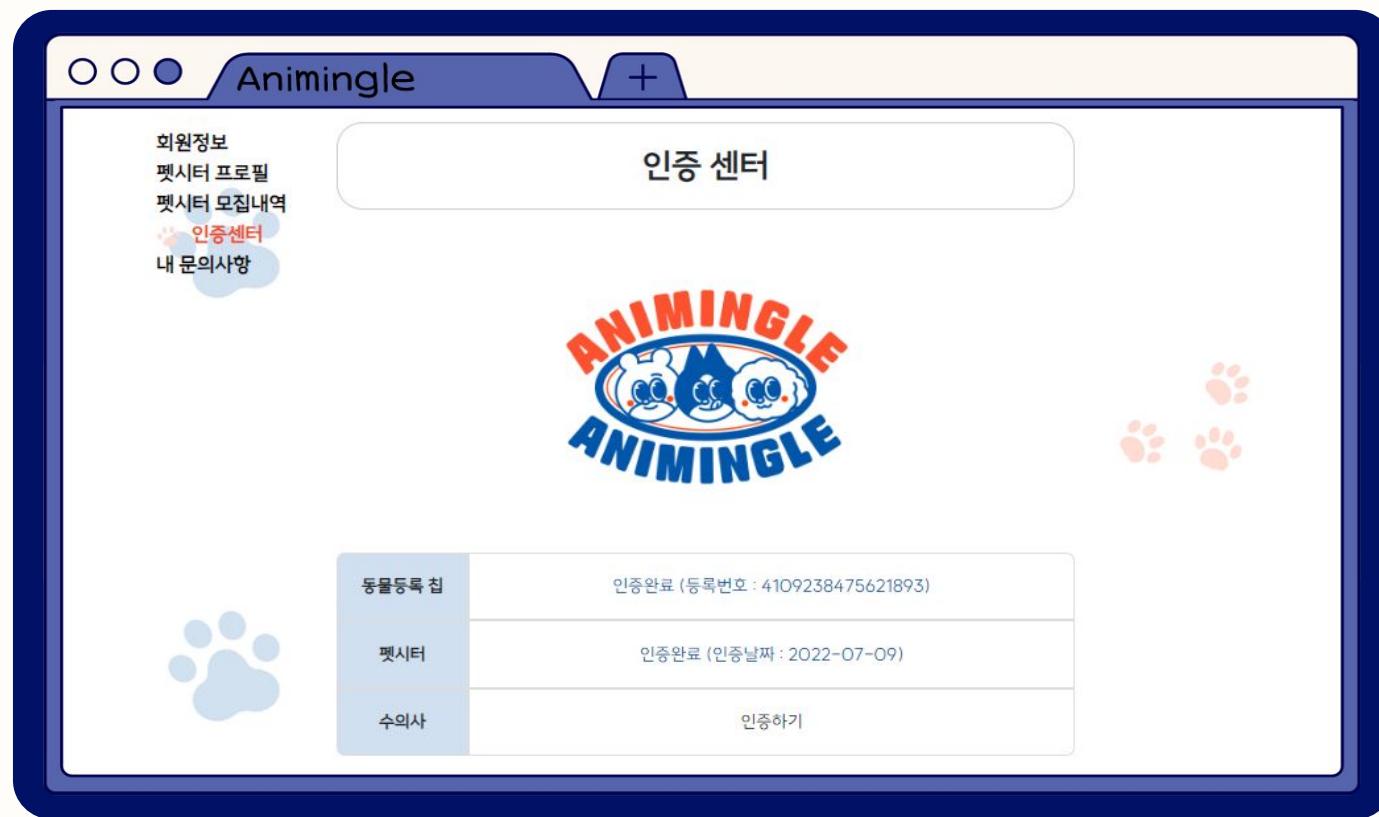
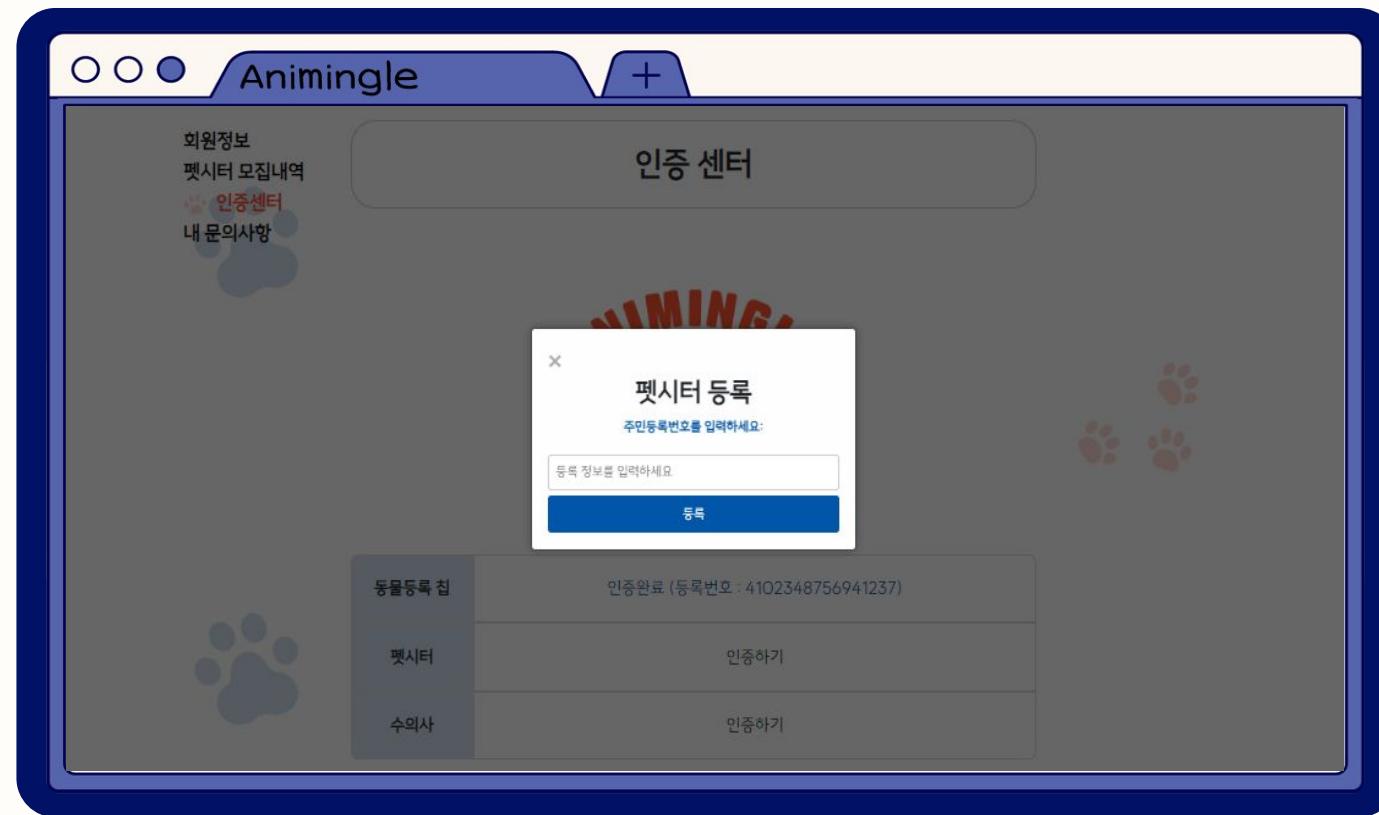
    private static Set<Session> clients = Collections.synchronizedSet(new HashSet<Session>());
}

@OnMessage
public void onMessage(String message, Session session) throws IOException {
    System.out.println(message);
    synchronized(clients) {
        for(Session client : clients) {
            if(!client.equals(session)) {
                client.getBasicRemote().sendText(message);
            }
        }
    }
}

@OnOpen
public void onOpen(Session session) {
    System.out.println(session);
    clients.add(session);
}

@OnClose
public void onClose(Session session) {
    clients.remove(session);
}
}
```

마이페이지



```
/* 동물 등록 클릭 시 */
$(document).ready(function() {
  $('#auth-animal').click(function() {
    openModal('동물 등록', '동물 등록 번호를 입력하세요:');
  });

  /* 펫시터 등록 클릭 시 */
  $('#auth-petsitter').click(function() {
    openModal('펫시터 등록', '주민등록번호를 입력하세요:');
  });

  $('#auth-doctor').click(function() {
    openModal('수의사 등록', '수의사 등록 번호를 입력하세요:');
  });
});

function openModal(title, description) {
  $('#modal-title').text(title);
  $('#modal-description').text(description);
  $('#myModal').css('display', 'block');

  if (title == '동물 등록') {
    $('#registration-type').val('pet');
  } else if (title == '펫시터 등록') {
    $('#registration-type').val('petsitter');
  } else if (title == '수의사 등록') {
    $('#registration-type').val('vet');
  }
}
```

마이페이지

The screenshot shows the '회원 정보' (User Information) section of the Animingle user profile. It includes fields for name (최유안), gender (여자), address (서울특별시 서초구 방배로13길 40), phone number (594-1522), and birthdate (010-8250-1152). There are also fields for old password (현재 비밀번호), new password (새 비밀번호), and confirmation (새 비밀번호 확인). A '수정' (Update) button is at the bottom.

The screenshot shows the '수의사 프로필' (Veterinarian Profile) section of the Animingle user profile. It includes fields for name (박우현), hospital name (24시 애니동물병원), address (서울특별시 강북구 보문로 99 (보문동5가)), and specialty (영양분석). There are also fields for degree (서울대 수의학과 대학원 석사), additional degree (진로과목), and certificate (파종증). A '수정' (Update) button is at the bottom. At the bottom of the page, there is a '답변 내역' (Reply History) section with a message from '노즈워크' (Nozwork) dated '2023-06-06'.

```
$( '#editpic' ).change(function() {  
    editPic();  
});  
  
function editPic() {  
  
    const formData = new FormData(document.getElementById('formpic'));  
  
    $.ajax ({  
        type: "POST",  
        url: '/animingle/user/profilepicedit.do',  
        enctype: 'multipart/form-data',  
        processData: false,  
        contentType: false,  
  
        data: formData,  
  
        dataType: 'json',  
        success: (result)=>{  
  
            $('#userpic').attr("src","/animingle/asset/pic/userpic/" + result.userpic);  
            $('#oldpic').val(result.userpic);  
            alert('사진 수정이 완료되었습니다.');  
  
        },  
        error: (a,b,c)=>console.log(a,b,c)  
    });  
}
```

마이페이지



```
<div class="content-top">
    <div class="content-title">펫시터 모집내역</div>
</div>

<div class="pr-content">
    
    <table class="tblcontent">
        <thead>
            <tr>
                <th>내 펫시터 모집내역</th>
            </tr>
        </thead>
        <tbody>
            <c:forEach items="${psrlist }" var="psrdto">
                <tr onclick="location.href='/animingle/board/petsitterview.do?psr_seq=${psrdto.psr_seq}&page=1'" style="cursor: pointer;">
                    <td>
                        <%-- <div class="date">[${psrdto.psr_regdate }]</div> --%>
                        <div class="subject-box">${psrdto.psr_subject}</div>
                        <div>
                            <c:if test="${psrdto.psr_state == '모집완료'}">
                                <div class="state-complete">${psrdto.psr_state}</div>
                            </c:if>
                            <c:if test="${psrdto.psr_state == '모집중'}">
                                <div class="state-ing">${psrdto.psr_state}</div>
                            </c:if>
                            <small class="date">${psrdto.psr_regdate}</small>
                        </div>
                    </td>
                </tr>
            </c:forEach>
            <c:if test="${psrlist.size() == 0 }">
                <tr style="cursor: default;">
                    <td style="display: table-cell; text-align: center;">신청 내역이 없습니다.</td>
                </tr>
            </c:if>
        </tbody>
    </table>
</div>
```

마이페이지

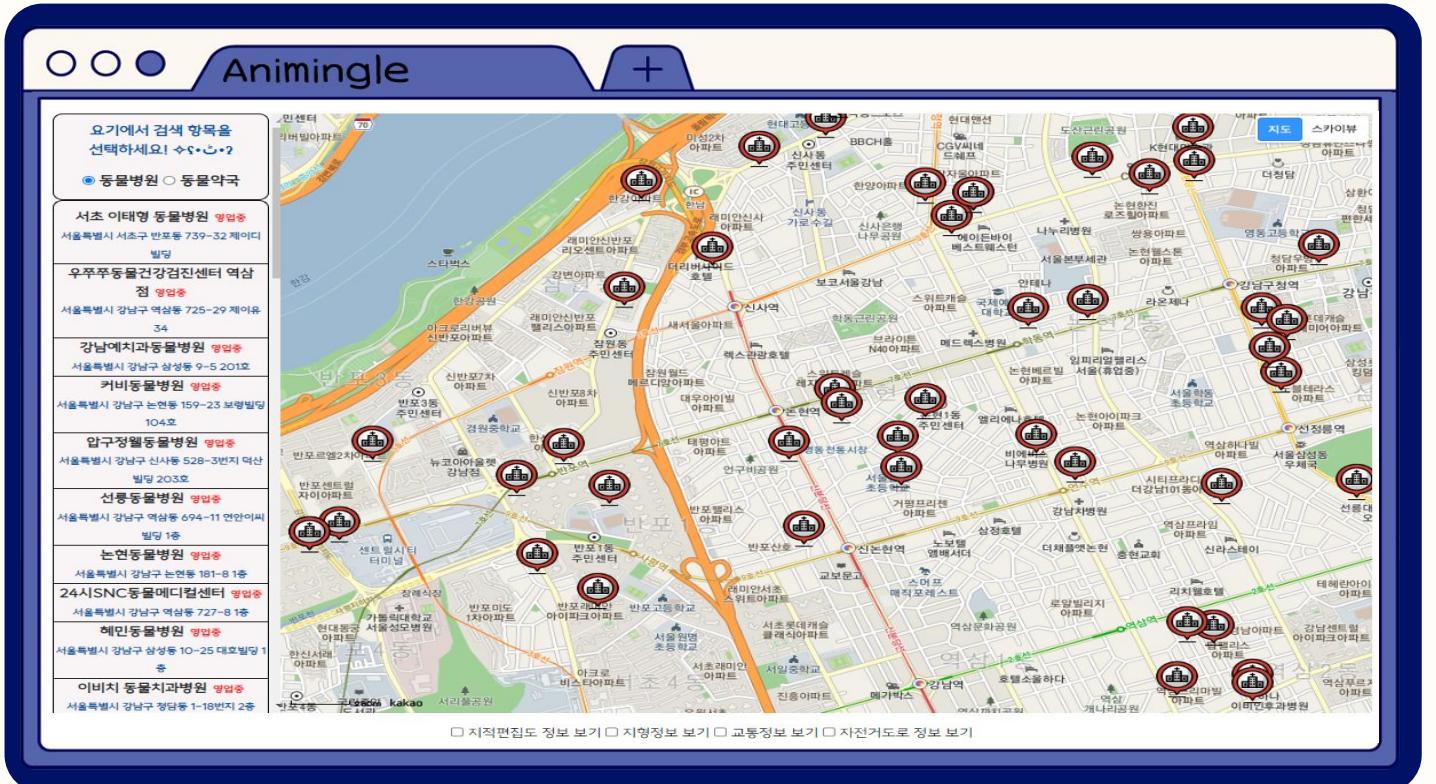
The screenshot shows the Animingle My Page interface. At the top, there's a navigation bar with icons for account settings, a plus sign, and a search bar containing 'Animingle'. Below the navigation is a sidebar with links: 회원정보 (Member Information), 펫시터 프로필 (Pet Sitter Profile), 펫시터 모집내역 (Pet Sitter Job Listings), 인증센터 (Authentication Center), and 내 문의사항 (My Consultations), which is highlighted with a red background.

The main content area features the 'ANIMINGLE' logo at the top. Below it is a section titled '내 문의사항' (My Consultations) with a light blue header. This section lists five consultation entries:

내 문의사항	답변완료	날짜
페이지가 로딩이 너무 느려요	답변완료	2022-06-09
페이지가 로딩이 너무 느려요	답변완료	2022-06-09
마이페이지가 이상해요	답변완료	2023-06-05
신고하고 싶어요	답변완료	2023-06-05
건의합니다	답변완료	2023-06-05

```
<tbody>
<c:forEach items="${mqlist }" var="mqdto" varStatus="status">
<c:if test="${mqdto.in_state == '답변완료' }">
<tr onclick="toggleAnswer('report${status.index}')" style="cursor: pointer;">
</c:if>
<c:if test="${mqdto.in_state == '확인중' }">
<tr>
</c:if>
<td>
${mqdto.in_subject }
<div>
<c:if test="${mqdto.in_state == '확인중' }">
<div class="state-ing">${mqdto.in_state }</div>
</c:if>
<c:if test="${mqdto.in_state == '답변완료' }">
<div class="state-complete">${mqdto.in_state }</div>
</c:if>
<small class="date">${mqdto.in_regdate }</small>
</div>
</td>
</tr>
<c:if test="${mqdto.in_state == '확인중' }">
<tr class="answer-row">
</c:if>
<c:if test="${mqdto.in_state == '답변완료' }">
<tr class="answer-row" id="report${status.index}" style="display: none;">
<td>
<div class="answer" style="width: 100%;">
<span class="material-symbols-outlined">subdirectory_arrow_right</span>
${mqdto.ina_content }
<div style="text-align: right;">${mqdto.a_nickname}(${mqdto.a_id})<small>
[${}{mqdto.ina_regdate }]</small></div>
</div>
</td>
</tr>
</c:if>
</c:forEach>
<c:if test="${mqlist.size() == 0 }">
<tr style="cursor: default;">
<td style="display: table-cell; text-align: center;">문의 내역이 없습니다.</td>
</tr>
</c:if>
</tbody>
```

병원, 약국 찾기 or 리뷰



```

function handleRadioButtonChange() {
    kakao.maps.event.addListener(map, 'dragend', function() {
        var bounds = map.getBounds(); // 현재 지도의 영역을 가져옵니다
        var sw = bounds.getSouthWest(); // 남서쪽
        var ne = bounds.getNorthEast(); // 북동쪽

        removeAllMarkers(); // 기존의 마커들을 모두 제거합니다

        var selectedValue = document.querySelector('input[name="location"]:checked').value;
        console.log("선택된 값:", selectedValue);

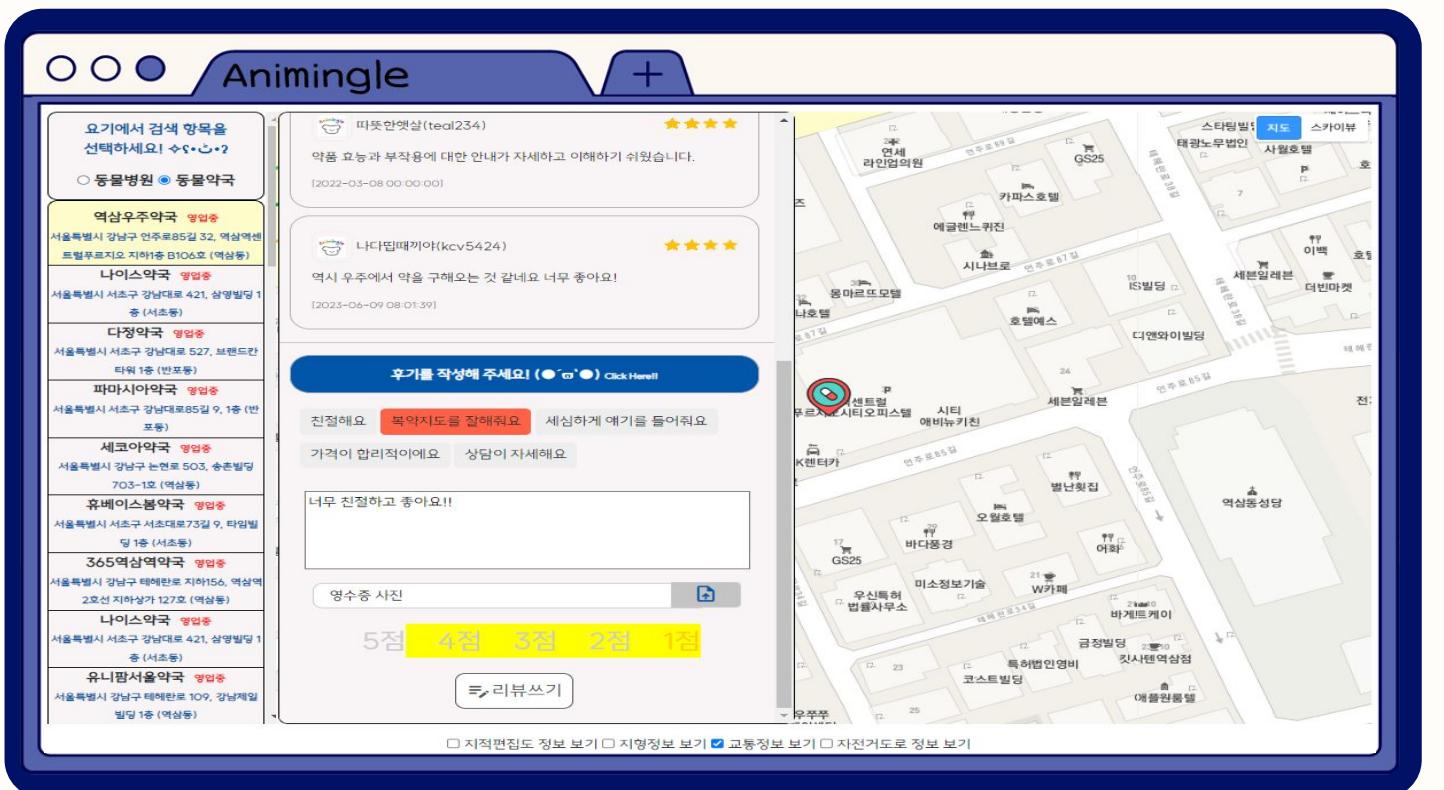
        globalVariable2 = selectedValue;
        if (selectedValue === "hospital") {
            console.log("동물병원 선택됨");
            // 동물병원에 대한 처리를 수행합니다
        }
    });
}

$.ajax({
    type: 'GET',
    url: '/animingle/board/medicalHospital.do',
    data: {
        neLat: ne.getLat(),
        neLng: ne.getLng(),
        swLat: sw.getLat(),
        swLng: sw.getLng(),
        select: "hospital"
    },
    dataType: 'json',
    success: function(result) {
        $("#side tbody").html("");
        createMarkersInBounds(result, bounds);
        console.log(result);
        if(result.length == 0){
            $("#side tbody").append(`<tr class="place-list">
                <th>현재 이곳에는 병원이 <br>존재하지 않아요<br><small id="small">&ampnbsp&ampnbsp↑(↑↑↑↑)</small><br>
                <small id="small2">다른 곳으로 이동해주세요 !!</small>
            </th>
            </tr>
        `);
        }
        $(result).each((index, item) => {
            //coords = new kakao.maps.LatLng(item.h_lat, item.h_lng);

            console.log('item', item.h_seq);
            $("#side tbody").append(`<tr class="place-list" id="${item.h_seq}" onclick="addside(${item.h_seq}, ${item.h_lat}, ${item.h_lng})">
                <th>${item.h_name}<small id="small">&ampnbsp&ampnbsp영업중</small><br>
                <small id="small2"><${item.h_address}</small>
            </th>
            </tr>
        `);
        });
        error: function(a,b,c) {
            console.log(a,b,c);
        }
    });
}

```

병원, 약국 찾기 or 리뷰



```

function addside(seq, lat, lng) {
    if ($('#addside').width() === 0) {
        globalVariable3 = seq;
        $.ajax({
            type: 'GET',
            url: '/animingle/board/medicalHReview.do',
            data: {
                seq: seq
            },
            dataType: 'json',
            success: function(result) {
                //result.arr1
                //result.arr2
                $(".review-list").html("");
                console.log("result : ", result);
                console.log("seq : ", seq);

                $('.title').text('장소');
                $('#address').text('주소');
                $('#time').text('오픈 시간');
                $('#call').text('닫는 시간');

                $(result.arr2).each(function(index, item) {
                    $('.title').text(item.h_name);
                    $('#address').text(item.h_address);
                    $('#time').text(item.h_open);
                    $('#call').text(item.h_close);

                    $(".review-list").append(`

                        <div class="item-wrap">
                            <div class="item-header">
                                <div>
                                     <span
                                        class="item-id">${item.userNickname}(${item.hr_writer})</span>
                                </div>
                                <div class="rating">
                                    ${generateStars(item.hr_rate)}
                                </div>
                            <div class="item-content">${item.hr_content}</div>

                            <div class="item-config">
                                <span>[${item.hr_regdate}]</span>
                            </div>
                        </div>
                    `);
                });
            }
        });
    }
}

```

로그인 ①펫시터, 수의사, 관리자 구분

SERVLET



```
//result: 일반회원과 관리자 구분해서 로그인
UserDTO user_result = dao.login(dto);
UserDTO admin_result = dao.login_admin(dto);

//1. 일반회원 로그인의 경우
if (user_result != null && admin_result ==null) {

    //1-1. 펫시터 체크
    int petsitter = dao.checkPetsitter(user_result.getUser_id());

    if (petsitter == 1) {
        req.getSession().setAttribute("isPet", "Y");
    }

    //1-2. 수의사 체크
    int veterinarian = dao.checkVet(user_result.getUser_id());

    if (veterinarian == 1) {
        req.getSession().setAttribute("isVet", "Y");
    }

    //3. 체크 후, 아이디, 닉네임 넣어주기
    req.getSession().setAttribute("id", user_result.getUser_id()); //인증 티켓 발급
    req.getSession().setAttribute("nickname", user_result.getUser_nickname());

    resp.sendRedirect("/animingle/index.do");
}

} else if (user_result == null && admin_result != null) { //2. 관리자 로그인의 경우

    req.getSession().setAttribute("id", admin_result.getUser_id());
    req.getSession().setAttribute("nickname", admin_result.getUser_nickname());
    req.getSession().setAttribute("isAdmin", admin_result.getIsAdmin());

    resp.sendRedirect("/animingle/index.do");
}

} else {
    //로그인 실패
    writer.print("<script>alert('로그인에 실패하였습니다.');" + history.back() + "</script>");
    writer.close();
}
```

로그인 ②소셜 로그인

JSP

```
//카카오 로그인
Kakao.init('aee326e29bebd9e80119e009577e3832');
function kakaoLogin() {
    Kakao.Auth.login({
        success: function (response) {
            Kakao.API.request({
                url: '/v2/user/me',
                success: function (response) {

                    //정보 받아오기
                    var data = {
                        id : response.id,
                        email : response.kakao_account.email,
                        nickname : response.kakao_account.profile.nickname,
                        profile_image :
                            response.kakao_account.profile.thumbnail_image_url,
                    };

                    $.ajax({
                        type : 'POST',
                        url : '/animingle/user/sociallogin.do',
                        data : data,
                        dataType : 'json',
                        success : result => {

                            if (result.result == "success") {
                                alert("로그인에 성공하였습니다.");
                                location.replace("/animingle/user/profile.do");
                            } else if (result.result == "failed") {
                                alert("로그인에 실패하였습니다. 다시 로그인 해주세요.");
                            }
                        },
                        error : (a,b,c) => console.log(a,b,c)
                    });
                },
                fail: function (error) {
                    alert(JSON.stringify(error));
                }
            },
            fail: function (error) {
                alert(JSON.stringify(error));
            }
        })
    }
}
```

SERVLET

```
//카카오로 로그인한 유저가 유저 테이블에 존재하는지?
int result = dao.kakaoUserCheck(dto);

resp.setCharacterEncoding("UTF-8");
resp.setContentType("application/json");

PrintWriter writer = resp.getWriter();

JSONObject obj = new JSONObject();

//1. 유저가 있다! => 로그인
if (result == 1) {

    UserDTO result1 = dao.kakaoUserLogin(dto);
    //로그인 성공시
    if (result1 != null) {
        req.getSession().setAttribute("id", result1.getUser_id()); //인증 티켓 발급
        req.getSession().setAttribute("nickname", result1.getUserNickname());
        req.getSession().setAttribute("socialLogin", "kakao");
        req.getSession().setAttribute("isVet", null);
        req.getSession().setAttribute("isPet", null);
        req.getSession().setAttribute("isAdmin", null);

        obj.put("result", "success");
        writer.print(obj);
        writer.close();
    } else {
        obj.put("result", "failed");
        writer.print(obj);
        writer.close();
    }
} else if (result == 0) { //2. 유저가 없다! => user 테이블에 정보 넣고 로그인

    int result2 = dao.kakaoUserJoin(dto); //user 테이블에 정보 넣기

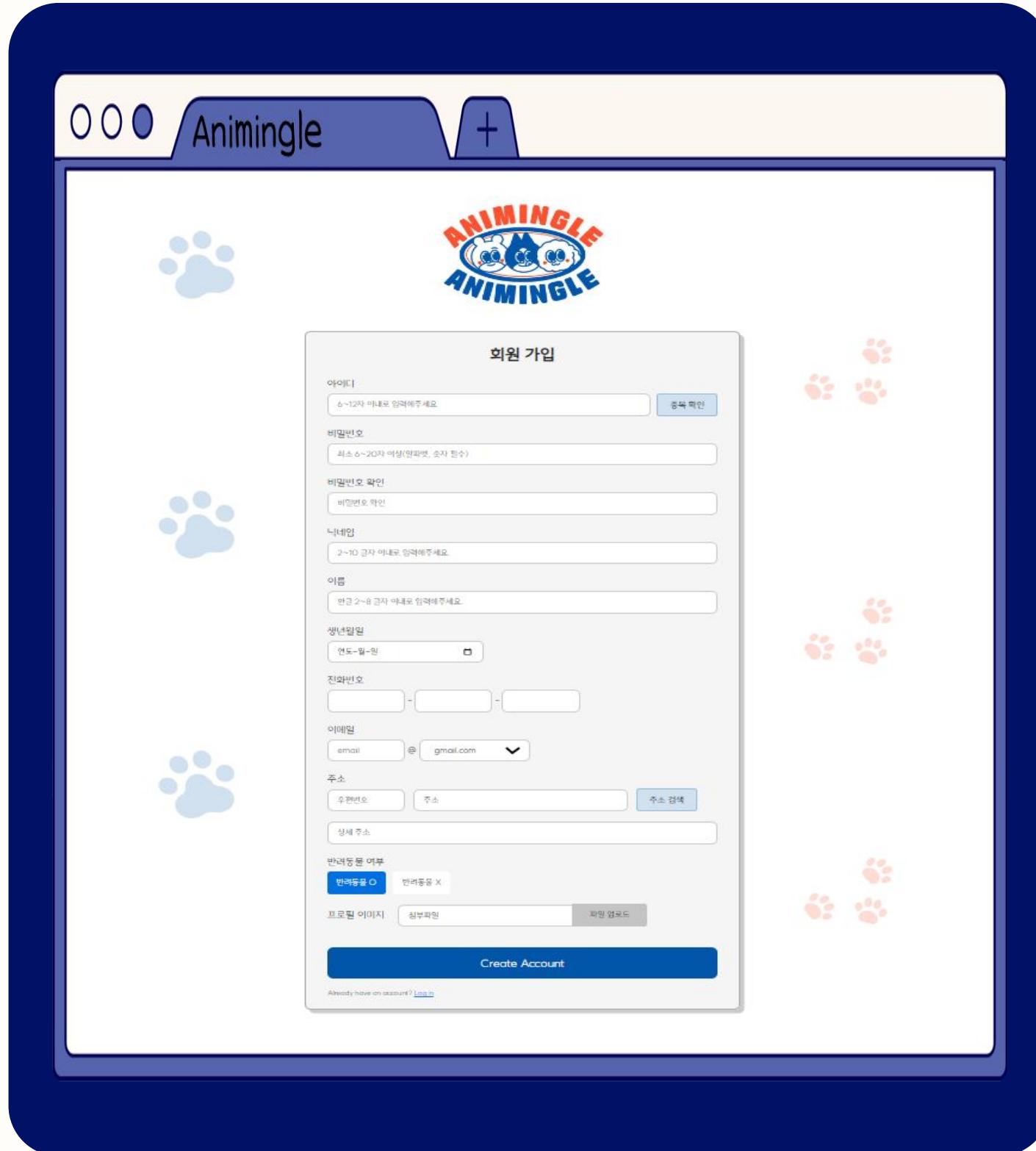
    if (result2 == 1) {
        //정보넣기 성공시
        UserDTO result3 = dao.kakaoUserLogin(dto); //user 로그인

        if (result3 != null) {
            //로그인 성공시
            req.getSession().setAttribute("id", result3.getUser_id()); //인증 티켓 발급
            ...

            obj.put("result", "success");
            writer.print(obj);
            writer.close();
        } else {
            obj.put("result", "failed");
            writer.print(obj);
            writer.close();
        }
    }
}
```

회원가입

JS



```
var checkValidate = {
    idcheck : false,
    idDuplicatecheck : false,
    pw1check : false,
    pw2check : false,
    nicknamecheck : false,
    namecheck : false,
    birthdaycheck : false,
    telok : false,
};
```

```
//유효성 검사 - 1. 아이디
var idRegExp = /^[a-zA-Z][a-zA-Z0-9]{5,11}$/;

$('.id').on('keyup', function() {
    var id = $('.id').val();

    if (!idRegExp.test(id)) {
        idresultDiv.textContent = "아이디는 영문 대소문자..";
        $('.id').css('outline', '1px solid #FC5230');
        checkValidate.idcheck = false;
        checkValidate.idDuplicatecheck = false;

        return false;
    } else {
        ...
        checkValidate.idcheck = true;
    }
});[])
```

```
//checkValidate 객체의 값들이 모두 true인지 확인하는 함수
function isAllTrue(obj) {
    for (var key in obj) {
        if (!obj[key]) {
            return false;
        }
    }
    return true;
}

function checkForm() {
    if (isAllTrue(checkValidate)) {
        return true;
    } else {
        if (checkValidate.idDuplicatecheck == false) {
            alert("아이디 중복검사를 해주세요.");
            $('.id').focus();
            event.preventDefault();
            return false;
        } else {
            alert("회원가입에 실패하였습니다.");
            $('.id').focus();
            event.preventDefault();
            return false;
        }
    }
}
```

고객센터



JSP

```
$('#add-btn').on('click', function() {  
    const inquiryContent = editor.getData();  
  
    $.ajax({  
        type: 'POST',  
        url: '/animingle/board/inquiry.do',  
        data: {  
            prefix : $('#inquiry_brackets').val(),  
            title : $('#title').val(),  
            content : inquiryContent  
        },  
        dataType: 'json',  
        success: function (result) {  
            if (result.result == 1) {  
                alert("문의가 성공적으로 등록되었습니다.");  
                window.location = '/animingle/index.do';  
            } else {  
                alert("문의 등록에 실패하였습니다.");  
            }  
        },  
        error: (a,b,c)=>console.log(a,b,c)  
    });  
});
```

SERVLET

```
String prefix = req.getParameter("prefix");  
String title = req.getParameter("title");  
String content = req.getParameter("content");  
  
HttpSession session = req.getSession();  
String id = (String)session.getAttribute("id");  
  
InquiryDTO dto = new InquiryDTO();  
dto.setIn_prefix(prefix);  
dto.setIn_subject(title);  
dto.setIn_content(content);  
dto.setIn_writer(id);  
  
//날짜날기  
Calendar now = Calendar.getInstance();  
dto.setIn_regdate(String.format("%tF\n", now));  
  
InquiryDAO dao = new InquiryDAO();  
  
int result = dao.addInquiry(dto);  
  
//json으로 결과보내기  
resp.setCharacterEncoding("UTF-8");  
resp.setContentType("application/json");  
  
PrintWriter writer = resp.getWriter();  
  
JSONObject obj = new JSONObject();  
obj.put("result", result);  
writer.print(obj);  
writer.close();
```

관리 센터

000 Animingle +

처리현황

문의현황 : 10	문의완료 : 5	신고현황 : 12	신고완료 : 11
-----------	----------	-----------	-----------

문의 내역 그래프

일자별 요약

일자	문의	신고
2022-06-09	-	1
2022-09-12	-	1
2022-11-14	-	1
2023-01-05	-	1
2023-04-11	-	1
2023-05-18	-	1
2023-05-31	-	1
2023-06-05	4	3
2023-06-08	1	1
2023-06-09	5	1

```
$(document).ready(function() {
    var label = [];
    <c:forEach items="${graphCnt}" var="dto">
        label.push('${dto.in_regdate}');
    </c:forEach>

    var data3 = [];
    <c:forEach items="${graphCnt}" var="dto">
        data3.push('${dto.cnt2}');
    </c:forEach>

    var ctx =
        document.getElementById('wmgbatt').getContext('2d');
        type: 'line',
        data: {
            labels: label,
            datasets: [
                {
                    label: '문의 현황',
                    data: data2,
                    backgroundColor: [
                        'rgba(255, 99, 132, 0.2)',
                        'rgba(54, 162, 235, 0.2)',
                        'rgba(255, 206, 86, 0.2)',
                        'rgba(75, 192, 192, 0.2)',
                        'rgba(153, 102, 255, 0.2)',
                        'rgba(255, 159, 64, 0.2)'
                    ],
                    borderColor: [
                        'rgba(255, 99, 132, 1)',
                        'rgba(54, 162, 235, 1)',
                        'rgba(255, 206, 86, 1)',
                        'rgba(75, 192, 192, 1)',
                        'rgba(153, 102, 255, 1)',
                        'rgba(255, 159, 64, 1)'
                    ],
                    borderWidth: 1
                },
                {
                    label: '신고 현황',
                    data: data3,
                    backgroundColor: 'rgba(255, 205, 86, 0.2)',
                    borderColor: 'rgba(255, 205, 86, 1)',
                    borderWidth: 1
                }
            ],
            options: {
                scales: {
                    y: {
                        beginAtZero: true
                    }
                },
                maintainAspectRatio: false
            }
        });
});
```

관리 센터



JSP

```
$('#add-btn').on('click', function () {
    const answerContent = editor.getData();

    $.ajax({
        type: 'POST',
        url: '/animingle/user/admin/inquiryview.do',
        data: {
            in_seq : ${indto.in_seq},
            content : answerContent
        },
        dataType: 'json',
        success: function (result) {
            if (result.result == 1) {
                //답변 추가
                let div = `
                    <div class="answer">
                        <div class="answer-config">
                            <span>${result.a_id}(${result.a_nickname})</span>
                        </div>
                        <div>${result.content}</div>
                        <div>${result.regdate}</div>
                    </div>
                `;
                $('.answer-list').append(div);
                editor.setData("");
            } else {
                alert("답변 등록에 실패하였습니다.")
            }
        },
        error: (a,b,c)=>console.log(a,b,c)
    });
});[]}
```

SERVLET

```
InAnswerDTO dto = new InAnswerDTO();

dto.setA_id((String)
    HttpSession session.getAttribute("id"));
dto.setIna_content(content);
dto.setIna_regdate(String.format("%tF", now));

InAnswerDAO dao = new InAnswerDAO();

int result = dao.addAnswer(dto);
//새로 등록한 답변 내용 불러오기
InAnswerDTO newAnswer = dao.getAnswer(seq);

//json으로 결과보내기
resp.setCharacterEncoding("UTF-8");
resp.setContentType("application/json");

PrintWriter writer = resp.getWriter();

JSONObject obj = new JSONObject();
obj.put("result", result);
obj.put("content", newAnswer.getIna_content());
obj.put("a_id", newAnswer.getA_id());
obj.put("regdate", newAnswer.getIna_regdate());
obj.put("a_nickname", newAnswer.getA_nickname());

writer.print(obj);
writer.close();
```

커뮤니티 보드

산책 친구 구해요 >

와글와글 >

수의사 Q&A >

하늘다람쥐(wain1719) 2023-06-09
[나눔] 개껌 나눔해요!
⑦ 76 0

기타 30 0
QnA Test pineappe4 2023-06-15

고요한새벽(fuchsia678) 2023-06-05
[일상] 밥그릇 나눔해요!!
⑦ 51 2

강아지 6 0
강아지가 아파요 pecan789 2023-06-10

청량한물결(cornflower234) 2023-06-05
[나눔] 사료 나눔해요!
⑦ 14 1

강아지 19 6
노즈워크를 추천드립니다. seohyun05 2023-06-06

소중한추억(tangerine012) 2023-06-05
[나눔] 도넛 방석 나눔해요!
⑦ 40 4

고양이 19 0
QnA Test happy012 2023-05-15

```
<c:forEach items="${waglelist}" var="dto">
    <div class="content-list-box" onclick="location.href='/animingle/board/wagleview.do?seq=${dto.wg_seq}';">
        <div class="user-info">
            
            <span>${dto.wg_nickname}(${dto.wg_writer})</span>
            <span>${dto.wg_regdate}</span>
        </div>
        <div class="content-list-title wagletitle">
            <span>[${dto.wg_prefix}]</span>
            <span>${dto.wg_subject}</span>
        </div>
        <div class="content-views">
            <span class="material-symbols-outlined">visibility</span>
            <span class="views-count">${dto.wg_readcount}</span>
            <span class="material-symbols-outlined">chat</span>
            <span class="views-count">${dto.wg_ccnt}</span>
        </div>
    </div>
</c:forEach>
```

```
<c:forEach items="${qnalist}" var="dto">
    <div class="content-list-box vqnabox" onclick="location.href='/animingle/board/vetqnaview.do?vq_seq=${dto.vq_seq}';">
        <div class="vqna-top">
            <div class="tag">
                <c:if test="${dto.vq_prefix == '강아지'}">
                    
                </c:if>
                <c:if test="${dto.vq_prefix == '고양이'}">
                    
                </c:if>
                <span>${dto.vq_prefix}</span>
            </div>
            <div class="content-views">
                <span class="material-symbols-outlined">visibility</span>
                <span class="views-count">${dto.vq_readcount}</span>
                <span class="material-symbols-outlined">chat</span>
                <span class="views-count">${dto.answer_cnt}</span>
            </div>
        </div>
        <div class="content-list-title vqnatitle">
            <span>${dto.vq_subject}</span>
        </div>
        <div class="sub-content vqnasub">
            <div>
                ${dto.vq_writer}
            </div>
            <div>
                ${dto.vq_regdate}
            </div>
        </div>
    </div>
</c:forEach>
```

커뮤니티 - 와글와글



```
int currentPage = 1; // 현재 페이지 번호
int itemsPerPage = 5; // 페이지당 항목 수

String pageParam = req.getParameter("page");
if (pageParam != null) {
    currentPage = Integer.parseInt(pageParam);
}

if ((searchtext == null) || searchtext.equals("")) {
    search = "n";
} else {
    search = "y";
}

// 모든 게시물의 수를 담는다
int totalDataCount = dao.getTotalDataCount();

// 현재 페이지 수 계산
int totalPage = (int) Math.ceil((double) totalDataCount / itemsPerPage);

int maxPage = 5; // 한 번에 표시할 최대 페이지 수
int startPage = Math.max(currentPage - (maxPage / 2), 1);
int endPage = Math.min(startPage + maxPage - 1, totalPage);

map.put("searchtext", searchtext);
map.put("search", search);
map.put("currentPage", Integer.toString(currentPage));
map.put("itemsPerPage", Integer.toString(itemsPerPage));
map.put("prefix", prefix);

if (prefix == null) {
    List<WagleDTO> list = dao.list(map);

    for (WagleDTO dto : list) {
        dto.setWg_regdate(dto.getWg_regdate().substring(0, 10));
    }

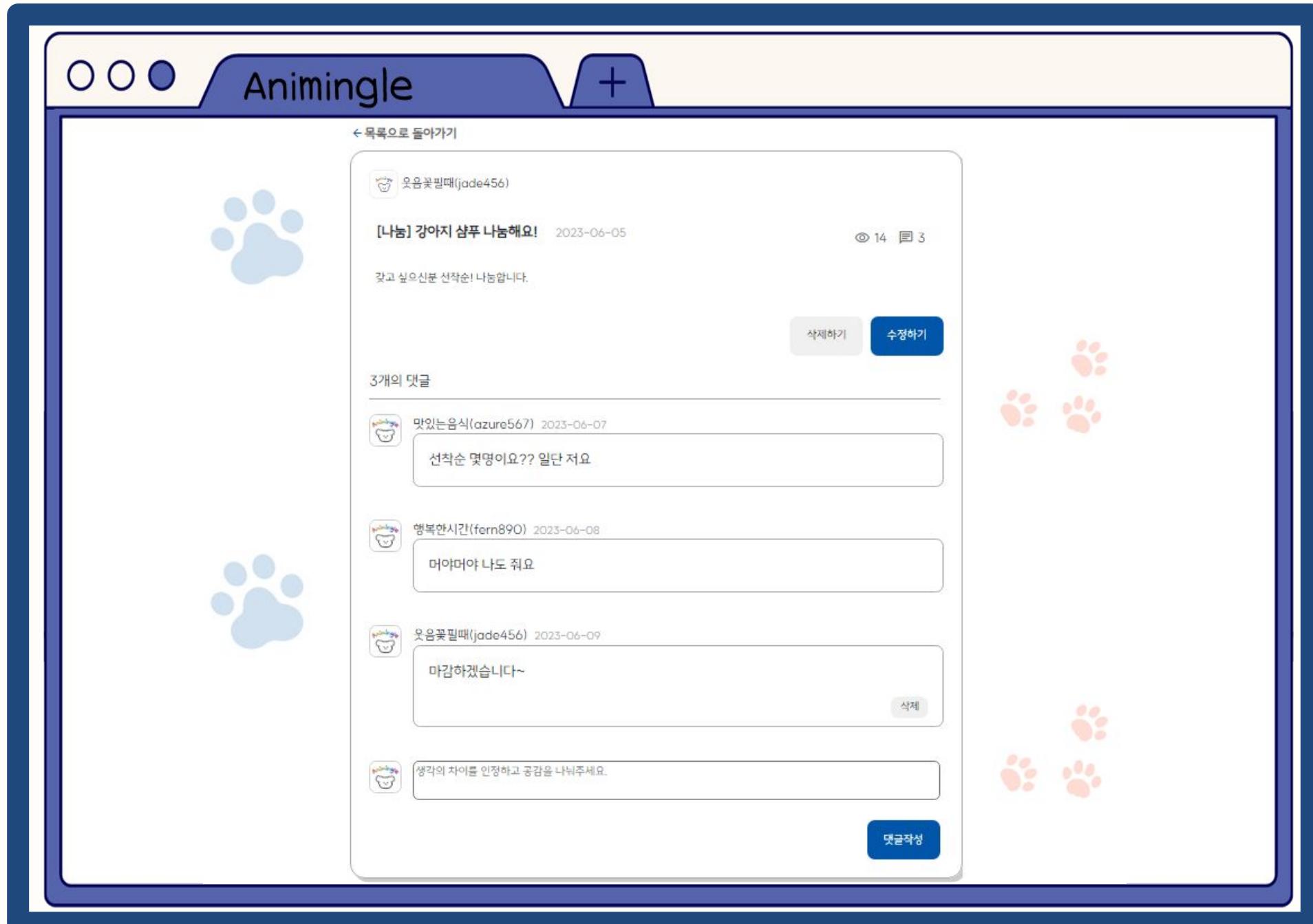
    req.setAttribute("list", list);
} else {
    List<WagleDTO> list = dao.prefixlist(map);

    for (WagleDTO dto : list) {
        dto.setWg_regdate(dto.getWg_regdate().substring(0, 10));
    }

    req.setAttribute("list", list);
}

req.setAttribute("currentPage", currentPage);
req.setAttribute("totalPage", totalPage);
req.setAttribute("startPage", startPage);
req.setAttribute("endPage", endPage);
req.setAttribute("searchtext", searchtext);
req.setAttribute("search", search);
req.setAttribute("prefix", prefix);
req.setAttribute("page", pageParam);
```

커뮤니티 - 와글와글



```
<div class="content-main">
    <span> ${dto.wg_content} </span>
</div>
<c:if test="${not empty id && (id == dto.wg_writer)}">
    <div class="btn-div">
        <button type="button" class="content-btn1"
               onclick="location.href='/animingle/board/wagleedit.do?seq=${dto.wg_seq}'>수정하기</button>
        <button type="button" class="content-btn2"
               onclick="location.href='/animingle/board/wagledel.do?seq=${dto.wg_seq}'>삭제하기</button>
    </div>
</c:if>
<div class="content-bottom">
    <div class="comment-top">
        <span>${map.ccnt}개의 댓글</span>
    </div>
    <c:forEach items="${clist}" var="cdto">
        <div class="comment-view">
            <div class="comment-image">
                
            </div>
            <div class="comment">
                <div class="comment-info">
                    <span>${cdto.wgc_nickname}(${cdto.wgc_writer})</span>
                    <span>${cdto.wgc_redate}</span>
                </div>
                <div class="comment-text">
                    ${cdto.wgc_content}
                    <c:if test="${not empty id && (id == cdto.wgc_writer)}">
                        <div class="comment-editdelbtn">
                            <button type="button" class="comment-delbtn"
                                   onclick="location.href='/animingle/board/wagledelcomment.do?cseq=${cdto.wgc_seq}&seq=${dto.wg_seq}'>삭제</button>
                        </div>
                    </c:if>
                </div>
            </div>
        </div>
    </c:forEach>
    <c:if test="${not empty id}">
        <form method="POST" action="/animingle/board/waglecommentadd.do">
            <div class="comment-add">
                <div class="comment-image">
                    
                </div>
                <div class="main-content-sel3">
                    <textarea name="content" id="editor"
                              placeholder="생각의 차이를 인정하고 공감을 나눠주세요.">
                </textarea>
            </div>
            <div class="comment-btn">
                <button type="submit" class="comment-btn1">댓글작성</button>
            </div>
            <input type="hidden" name="wg_seq" value="${dto.wg_seq}">
        </form>
    </c:if>
</div>
```

06

시연

THANK YOU

