

# TP 최종 보고서

201802161 조은빈

## 1. 실제로 구현한 코드의 라인 수

- \* CSS : 669
- \* JS : 1466
- \* PHP : 675
- \* HTML : 1610
- > Total : 4420

## 2. 구현 환경

- \* 개발 도구 : Atom
- \* 언어 : CSS, JS, PHP, HTML

## 3. 시나리오 및 코드 설명

### (1) 회원가입 - signUp.html

- \* 상단 메뉴바에서 '로그인'을 선택했을 때 나타나는 페이지이다.
- \* [그림 1]에서 '회원 가입' 버튼을 클릭하면 [그림 2]와 같이 아래에 회원 가입 정보 입력란이 나타난다.
- \* [그림 2]에서 '관리자로 가입' 버튼을 클릭하면 입력한 정보가 'mng\_signUp.php' 서버로 전송되고, '일반 회원으로 가입' 버튼을 클릭하면 'signUp.php' 서버로 전송된다.([그림 3] 참고)

LiveUs 로그인

회원인 경우 아이디와 비밀번호를 입력하고 로그인하세요.  
회원이 아닌 경우 '회원 가입' 버튼으로 회원 가입을 진행하세요.

아이디	<input type="text"/>
비밀번호	<input type="password"/>

로그인      회원가입

[그림 1]

**LiveUs 회원 가입 정보 입력**

직원 및 관리자의 경우에는 '관리자로 가입'으로 가입하고 일반인의 경우에는 '일반 회원으로 가입'으로 가입하세요.

이름	<input type="text"/> 한글로 5글자 이내로 입력
아이디	<input type="text"/> 영문, 숫자를 조합하여 6~15자 이내로 입력
비밀번호	<input type="password"/> 영문, 숫자, 특수기호를 조합하여 8~15자 이내로 입력
생년월일	2001년 1월 1일
휴대폰 번호	<input type="text"/> 010부터 시작 하이픈 없이 입력
이메일	<input type="text"/>

관리자로 가입 일반 회원으로 가입

[그림 2]

```
<div class="submit_btn_group">
  <input type="submit" value="관리자로 가입" formaction="http://localhost/텀프/source/php/mng_signup.php" formmethod="POST">
  <input type="submit" value="일반 회원으로 가입" formaction="http://localhost/텀프/source/php/signup.php" formmethod="POST">
</div>
</form>
```

[그림 3]

\* [그림 4]와 같이 Validation을 거친다.

**LiveUs 회원 가입 정보 입력**

직원 및 관리자의 경우에는 '관리자로 가입'으로 가입하고 일반인의 경우에는 '일반 회원으로 가입'으로 가입하세요.

이름	<input type="text"/> 한글로 5글자 이내로 입력
아이디	<input type="text"/> user@huh40 영문, 숫자를 조합하여 6~15자 이내로 입력
비밀번호	<input type="password"/> 영문, 숫자, 특수기호를 조합하여 8~15자 이내로 입력
생년월일	2001년 1월 1일
휴대폰 번호	<input type="text"/> 010부터 시작 하이픈 없이 입력
이메일	<input type="text"/> 이 입력란을 작성하세요. (john@grahner.com)

관리자로 가입 일반 회원으로 가입

[그림 4]

\* 회원 가입을 클릭하면 [그림 5]와 같이 알림창이 나타난다.

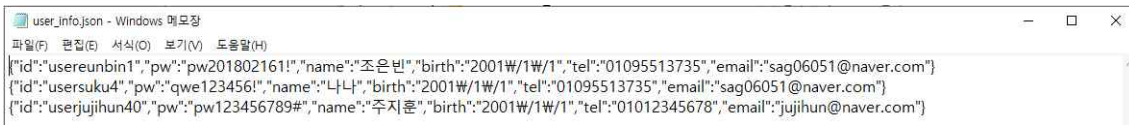
localhost 내용:

회원 가입이 완료되었습니다.

확인

[그림 5]

\* 일반 회원으로 가입했을 경우 user\_info.json 파일에 [그림 6]과 같이 정보가 저장된다.



[그림 6]

#### \* signUp.php

```

6  $user_name = $_POST['user_name'];
7  $user_id = $_POST['user_id'];
8  $user_pw = $_POST['user_pw'];
9  $user_birth_year = $_POST['user_birth_year'];
10 $user_birth_month = $_POST['user_birth_month'];
11 $user_birth_day = $_POST['user_birth_day'];
12 $user_tel = $_POST['user_tel'];
13 $user_email = $_POST['user_email'];
14 if(!isset($myObj)) $myObj = new stdClass();
15 $users = array();
16 $check = false;
17 if(!file_exists("../data/user_info.json")){ //user_info.json 파일이 없을 경우
18     $file = fopen("../data/user_info.json", "a+");
19     sign_up($file); // 파일을 생성하고 id, pw를 파일에 저장
20 }else{ //user_info.json 파일이 있을 경우
21     $file = fopen("../data/user_info.json", "a+");
22     while(!feof($file)){ // 파일 안의 json 문장들을 decode해서 users 배열에 삽입
23         $line = json_decode(fgets($file));
24         if($line != ""){
25             array_push($users, $line);
26         }
27     }

```

클라이언트가 POST 형식으로 보낸 이름, id, pw, 생년월일, 연락처, 이메일을 받는다. 일반 회원의 가입 정보를 저장하는 user\_info.json 파일이 없을 경우 파일을 생성하고 sign\_up 함수를 이용해 회원 가입 정보를 파일에 저장한다.

```

28 // id가 이미 존재하는지 검사
29 foreach ($users as $index => $mem) {
30     $in_id = $users[$index]->id;
31     if($in_id == $user_id){ // id가 존재할 경우
32         echo "<script>alert(\"이미 아이디가 존재합니다.\");</script>";
33         echo("<script>location.replace('../signUp.html');</script>");
34         $check = true;
35         break;
36     }
37 }
38 if(!$check){ // id가 존재하지 않을 경우
39     sign_up($file); // 회원가입
40 }
41 }
42 fclose($file);

```

파일이 있을 경우 파일 안의 내용들 중 id가 이미 존재하는지 검사한다. id가 존재하지 않으면 마찬가지로 sign\_up 함수를 이용해 회원 가입 정보를 파일에 저장한다. 만약 id가 존재하면 '이미 아이디가 존재합니다.'라는 알림창을 띄운 뒤 로그인 페이지로 돌아간다.

```

64 function sign_up($file){
65     global $user_id;
66     global $user_pw;
67     global $user_name;
68     global $user_birth_year;
69     global $user_birth_month;
70     global $user_birth_day;
71     global $user_tel;
72     global $user_email;
73     $user_birth = $user_birth_year."/".$user_birth_month."/".$user_birth_day;
74     $user = array("id"=>$user_id, "pw"=>$user_pw, "name"=>$user_name, "birth"=>$user_birth, "tel"=>$user_tel, "email"=>$user_email);
75     fwrite($file, json_encode($user, JSON_UNESCAPED_UNICODE)."\n"); // encode 해서 파일에 작성
76     echo "<script>alert(\"회원 가입이 완료되었습니다.\");</script>";
77     echo("<script>location.replace('../signUp.html');</script>");
78 }

```

파일에 저장한 후 ‘회원 가입이 완료되었습니다.’라는 알림창을 띄운 뒤 로그인 페이지로 돌아간다.

\* mng\_signUp.php

mng\_info.json 파일에 관리자 회원 정보를 저장한다. 동작 방식은 signUp.php와 같다.

## (2) 로그인 - signUp.html

\* 아이디 또는 비밀번호를 누락한 상태로 로그인 버튼을 클릭하면 [그림 10]과 같이 ‘아이디 또는 비밀번호를 입력하세요’ 알림창이 나타난다.

The screenshot shows a web interface for 'LiveUs 로그인'. At the top, there are navigation links: '센터소개' and '센터의 아이들'. A modal alert box is displayed in the center, titled 'localhost 내용:', with the message '아이디 또는 비밀번호를 입력하세요' and a '확인' button. Below the modal, the login form is visible. It has a title 'LiveUs 로그인' and instructions: '회원인 경우 아이디와 비밀번호를 입력하고 로그인하세요. 회원이 아닌 경우 '회원 가입' 버튼으로 회원 가입을 진행하세요.' The form contains two input fields: '아이디' (containing 'userjujihun40') and '비밀번호' (empty). At the bottom, there are two buttons: '로그인' (brown) and '회원가입' (orange).

[그림 10]

\* 아이디 또는 비밀번호가 일치하지 않을 경우 [그림 11]과 같이 알림창이 나타난다.

소개 센터의 아이들 로그인

LiveUs 로그인

회원인 경우 아이디와 비밀번호를 입력하고 로그인하세요.  
회원이 아닌 경우 '회원 가입' 버튼으로 회원 가입을 진행하세요.

아이디	userjujihun40
비밀번호	..

로그인 회원가입

[그림 11]

\* 로그인에 성공하면 [그림 12]와 같이 ‘(회원명)님 로그인 되었습니다’ 알림창이 나타난다.

소개 센터의 아이들 로그인

LiveUs 로그인

회원인 경우 아이디와 비밀번호를 입력하고 로그인하세요.  
회원이 아닌 경우 '회원 가입' 버튼으로 회원 가입을 진행하세요.

아이디	userjujihun40
비밀번호	*****

로그인 회원가입

[그림 12]

\* 로그인 되면 메뉴바의 ‘로그인’ 메뉴가 [그림 13]처럼 ‘마이페이지’로 바뀐다.



[그림 13]

### (3) 메인 화면 - main.html

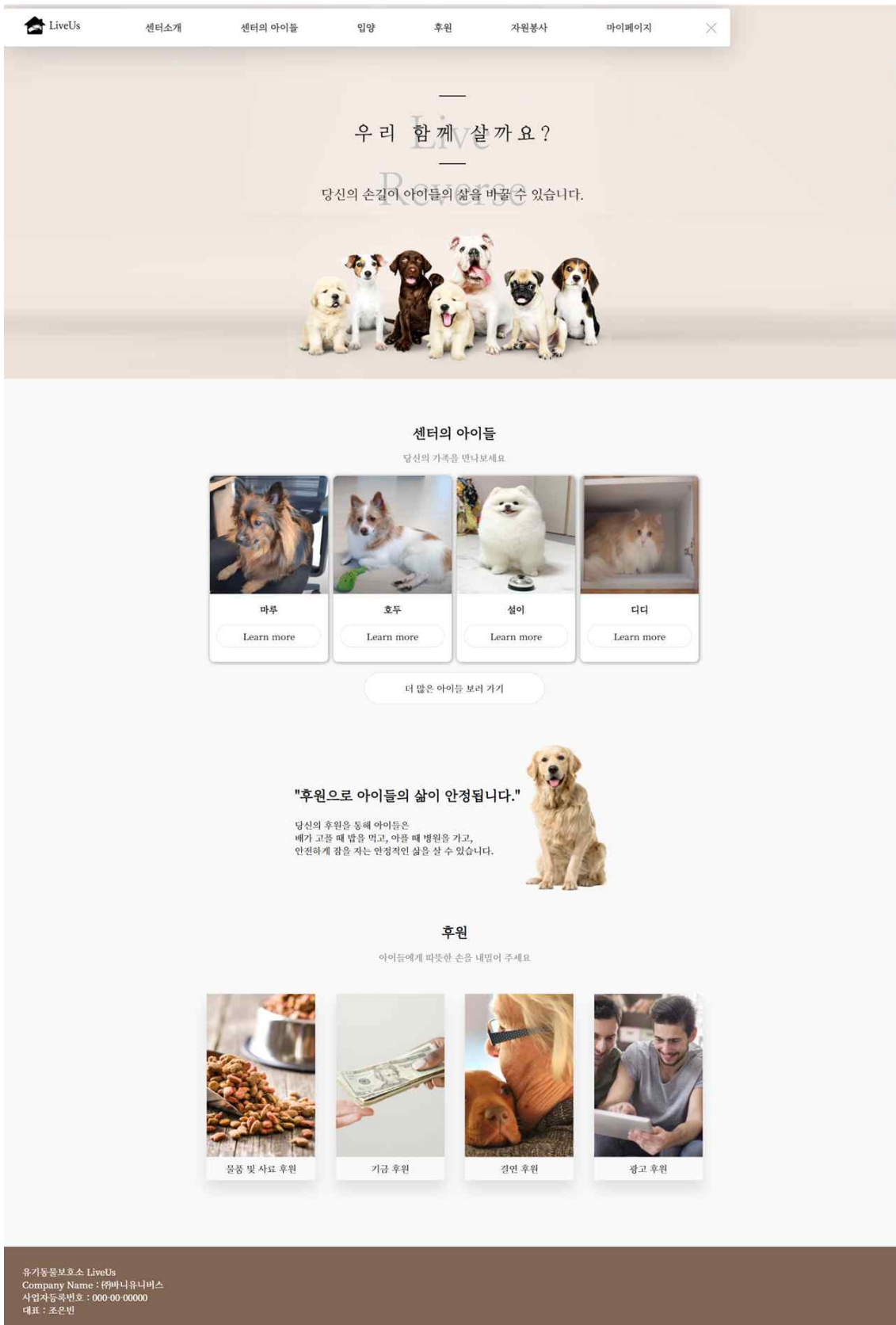
\* [그림 14]는 전체 메인 화면이다. 메인 화면은 메인 배너, 센터의 아이들 미리보기, 서브

카피, 후원 종류 설명 영역으로 나뉜다.

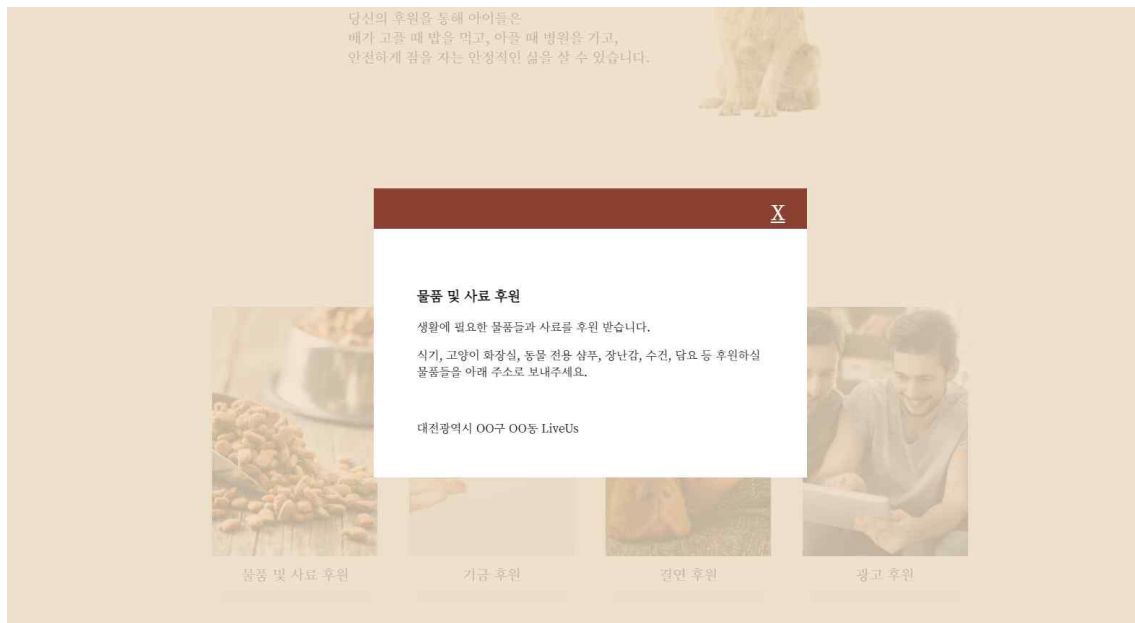
- \* 메인 배너를 클릭하면 '센터 소개' 페이지로 이동한다.

- \* 센터의 아이들 미리보기는 최대 4개까지 제공한다. '더 많은 아이들 보러가기' 버튼을 이용해 '센터의 아이들' 페이지로 이동한다.

- \* 후원 영역에 있는 이미지를 클릭하면 [그림 15]와 같이 후원 설명 팝업 레이어가 나타난다.



[그림 14]

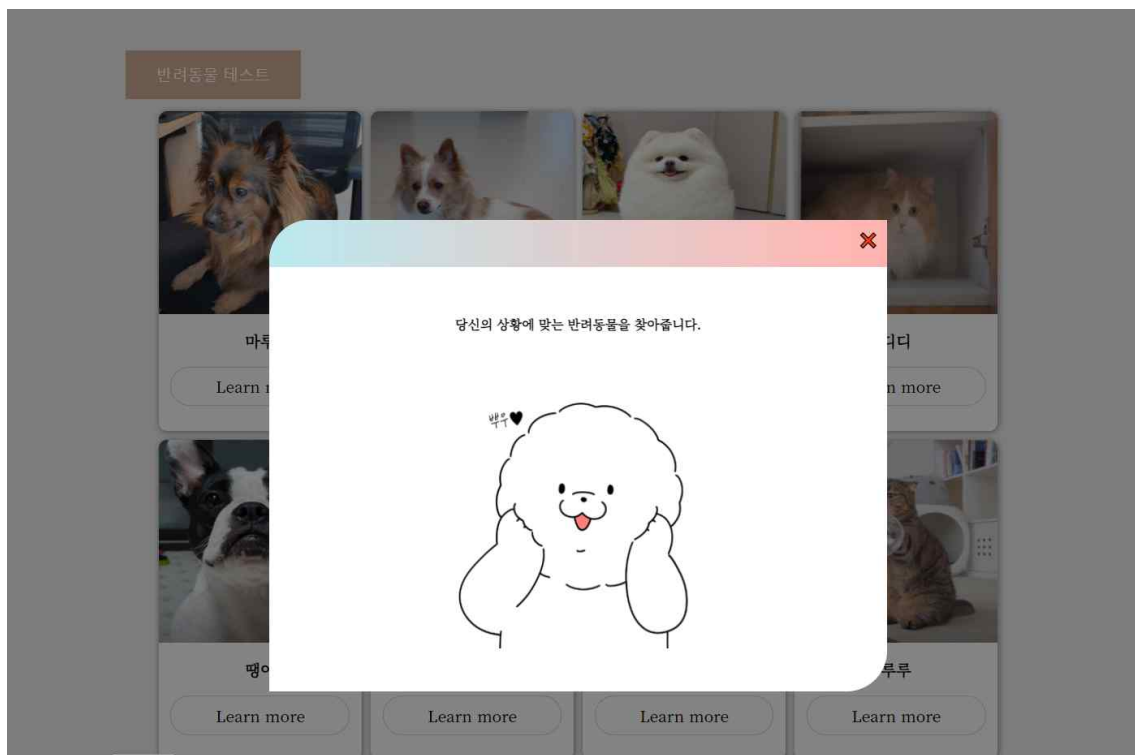


[그림 15]

#### (4) 일반 회원 시나리오 - children.html

##### ① 반려동물 테스트

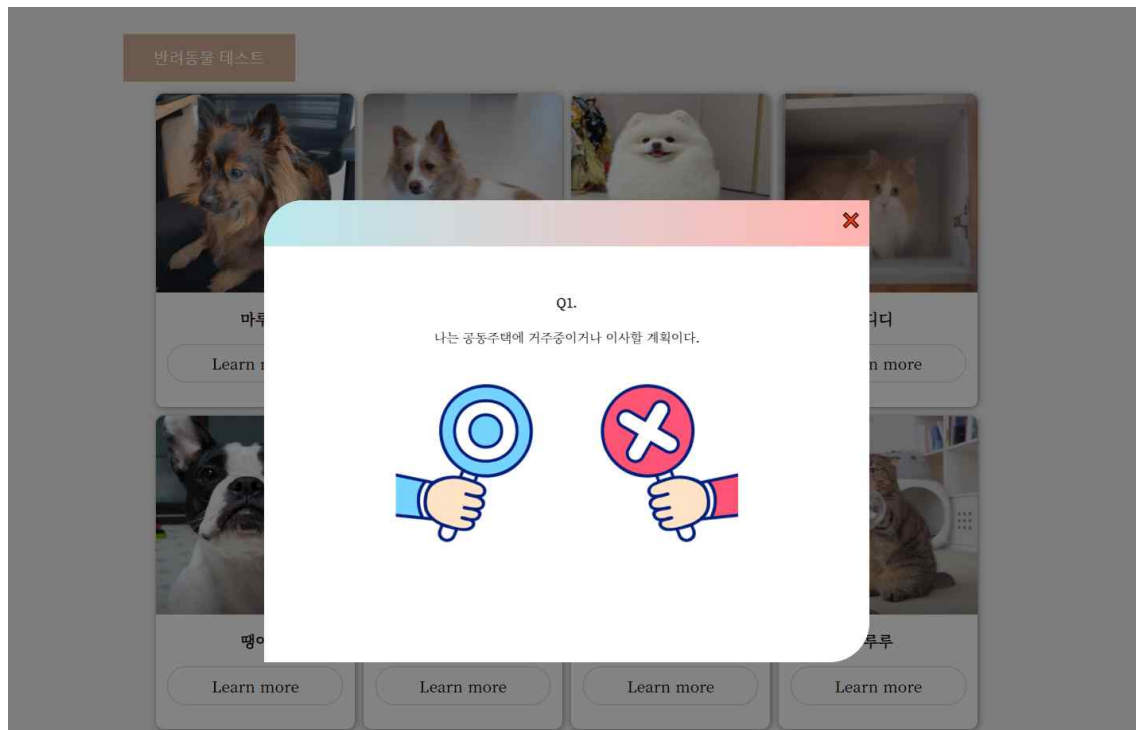
\* '센터의 아이들' 페이지에서 '반려동물 테스트'를 클릭하면 [그림 16]과 같이 테스트 레이어가 나타난다.



[그림 16]

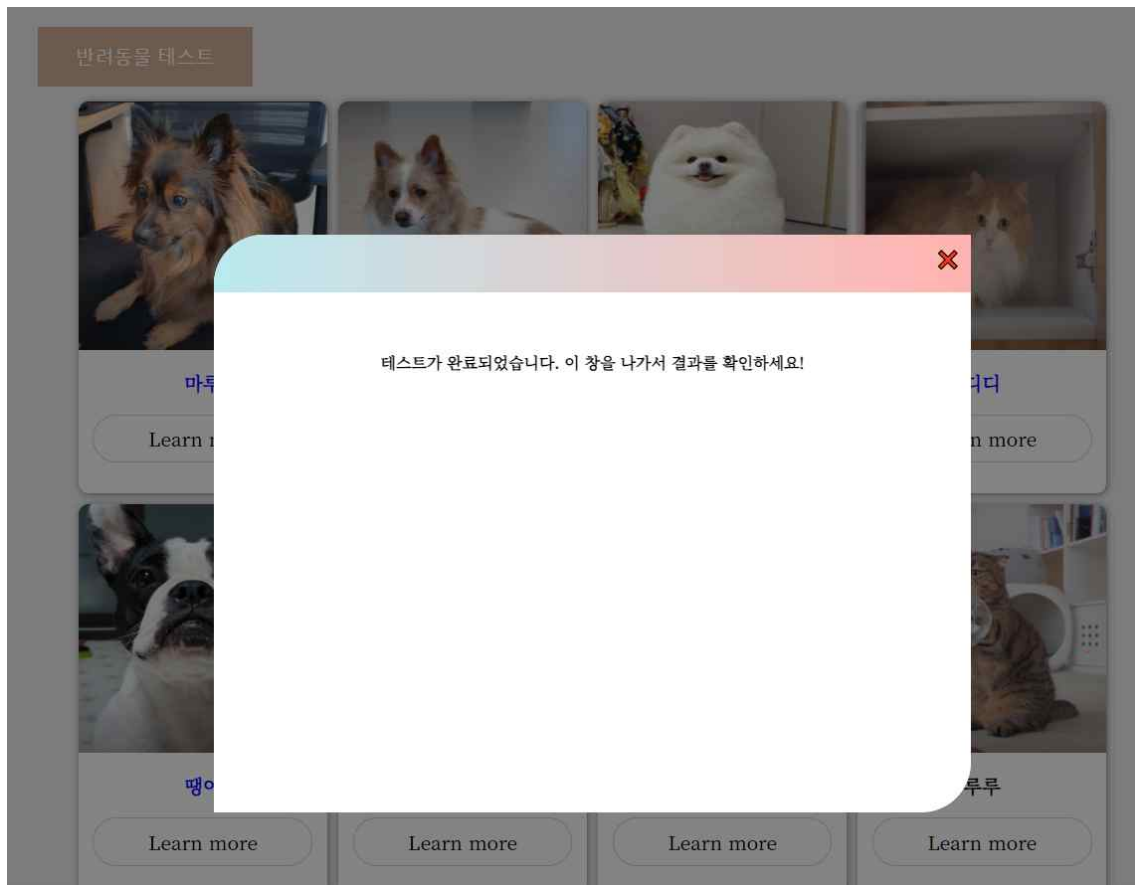


\* 이미지를 클릭하면 [그림 17]과 같이 테스트가 진행된다.



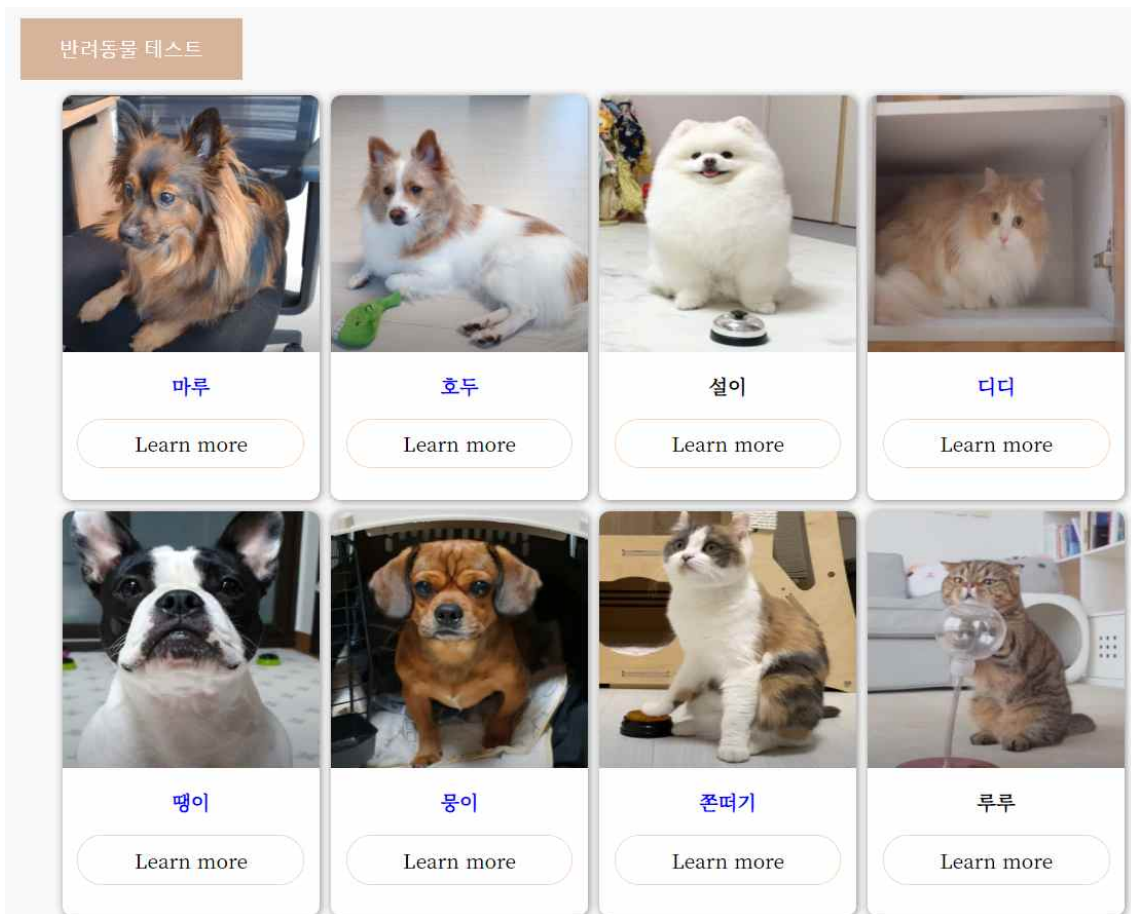
[그림 17]

\* [그림 18]은 테스트가 완료되었을 경우이다.



[그림 18]

\* 테스트 창을 나가면 [그림 19]와 같이 테스트 결과에 부합하는 아이들의 이름이 파란색으로 바뀐다.



[그림 19]

\* children.js

```

441 let current_question = 0; // 현재 질문 번호
442 let answers = [0, 0, 0, 0, 0, 0]; // 질문에 대한 답변에 따라 성격 카테고리 카운트
443
444 // 반려동물 테스트 질문 생성 및 결과 측정
445 function make_test() {
446   let test_content = document.getElementById('test_content');
447   switch (current_question) {
448     case 0: // 1번 질문
449       $('#question_num').text("Q1.");
450       $('#question').text("나는 공동주택에 거주중이거나 이사할 계획이다.");
451       $('#icon_o').off('click');
452       $('#icon_o').one('click', function() {
453         answers[2] += 1;
454         answers[4] += 1;
455         current_question += 1;
456         make_test();
457       });
458       break;

```

```
559 case 1: // 2번 질문
560     $('#question_num').text("Q2.");
561     $('#question').text("나는 재택근무를 한다.");
562     $('#icon_o').off('click');
563     $('#icon_o').one('click', function() {
564         answers[0] += 1;
565         answers[1] += 1;
566         answers[3] += 1;
567         answers[4] += 1;
568         current_question += 1;
569         make_test();
570     });
571     break;

574 case 8: // 테스트 종료
575     $('#question_num').text("테스트가 완료되었습니다. 이 창을 나가서 결과를 확인하세요!");
576     $('#question').text("");
577     $('#icon_o').hide();
578     $('#icon_x').hide();
579     show_test_result(answers);
580     break;
581 }
582 }

584 let children_set = []; // 모든 아이들 데이터 집합
585
586 // 테스트 결과 표시
587 function show_test_result(answers) {
588     var maxX = Math.max.apply(null, answers);
589     var index = answers.indexOf(maxX); // 가장 카운트 수가 높은 성격 항목의 인덱스
590     $('#card_list > li > div > p').css("color", "");
591     for(i in children_set){
592         for(j in children_set[i]['personality']){ // 아이들의 성격 중 해당하는 성격이 있으면 검색 결과로 나타냄
593             if(children_set[i]['personality'][j] == index + 1){
594                 let pet_name = children_set[i]['pet_name'];
595                 $('#'+pet_name).css("color", "blue");
596                 break;
597             }
598         }
599     }
600 }
601 }
```

current\_question으로 현재 질문 번호를 카운팅하고 질문에 대한 답변에 따라 성격 카테고리 카운팅해서 answers에 저장한다.

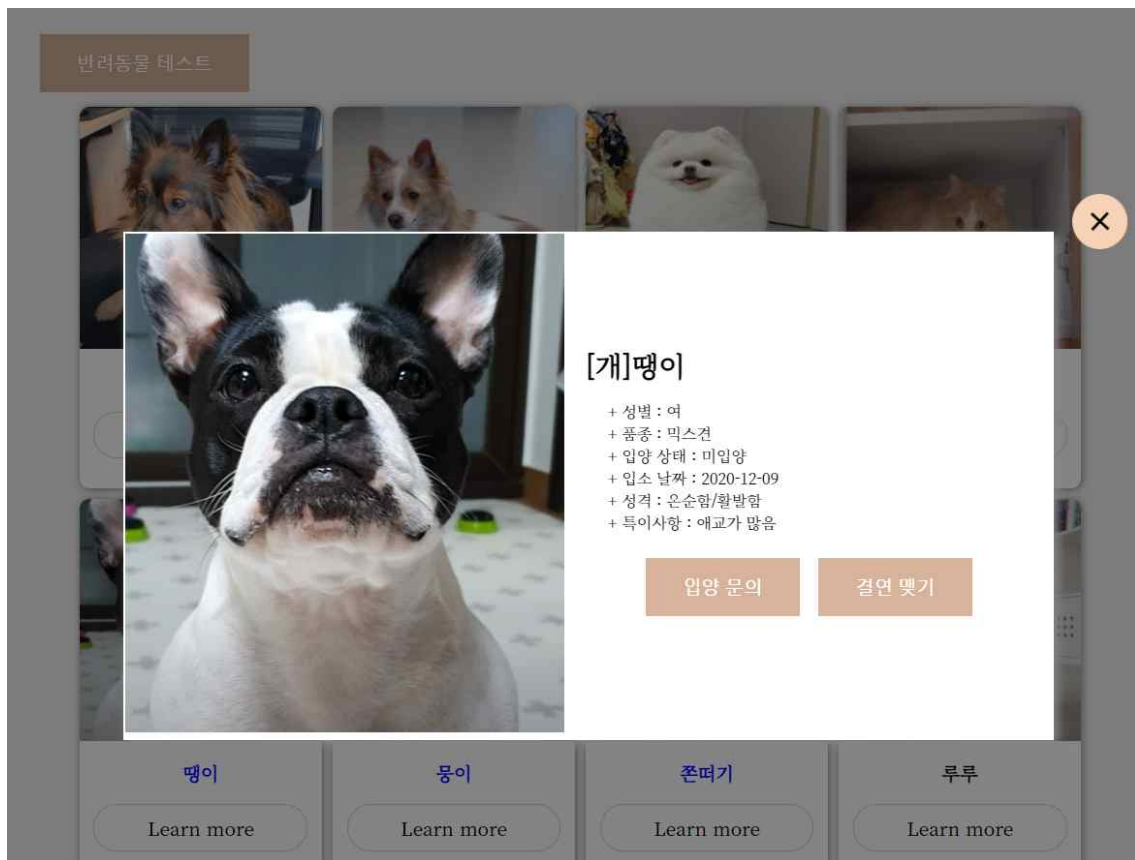
make\_test는 current\_question에 따라 질문을 생성하거나 결과를 측정한다. 질문에 따른 결과는 아래 표와 같이 분리하였다. 마지막 질문이 끝나면 show\_test\_result로 가장 O가 많은 성격 카테고리를 가진 아이를 결과로 출력한다.

	분리불 안	애정결 핍	온순함	사나움	조용함	활발함
1. 나는 공동주택에 거주중이 거나 이사할 계획이다.	x	x	o	x	o	x
2. 나는 재택근무를 한다.	o	o	x	o	o	x
3. 나는 어린 자녀가 있거나 자녀 계획이 있다.	o	x	o	x	o	o

4. 나는 사나운 개를 무서워한다.	X	O	O	X	O	X
5. 나는 반려동물이 이상행동이 있더라도 꾸준히 훈련시킬 인내심이 있다.	O	O	X	O	X	O
6. 나는 하루에 적어도 한 번은 산책을 나갈 수 있다.	X	X	X	O	X	O
7. 나는 경제적으로 부족함이 없다.	O	O	X	O	X	O
8. 나는 집에 이미 반려동물을 기르고 있다.	X	X	O	X	X	X

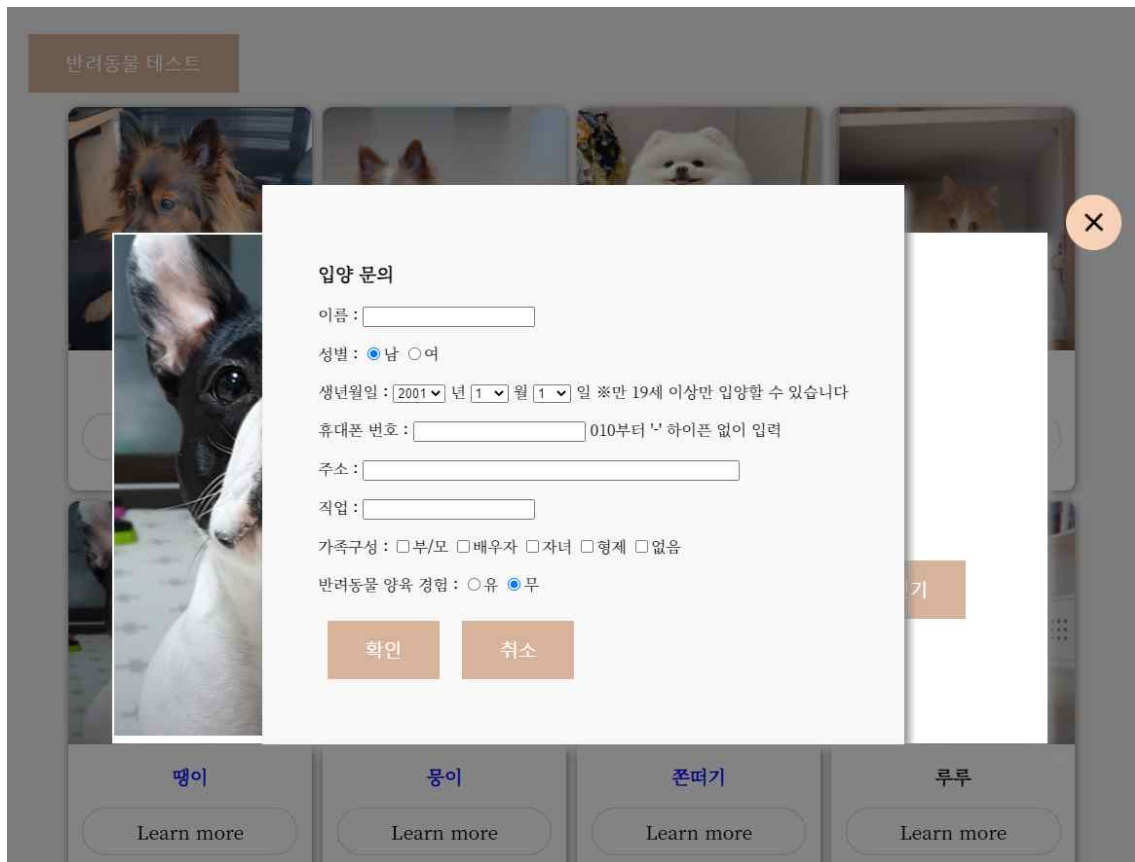
## ② 입양 문의

\* Learn more를 클릭하면 [그림 24]와 같이 디테일 카드가 나타난다.



[그림 24]

\* 입양 문의 버튼을 클릭하면 [그림 25]와 같이 입양 문의 레이어가 나타난다.

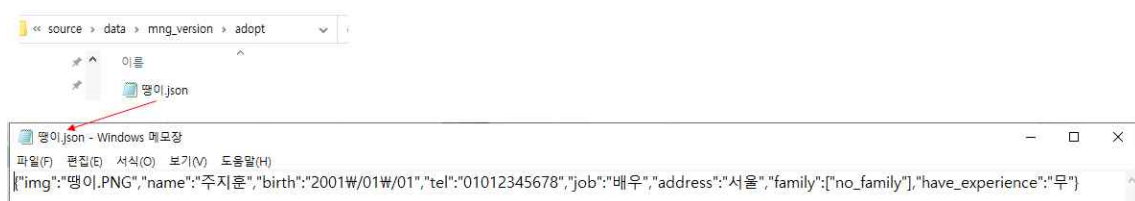


[그림 25]

\* 정보를 입력하고 확인 버튼을 누르면 user\_version/adopt에는 [그림 26]과 같이 입양 문의를 한 회원의 아이디로 파일이 생성된다. 파일 안에는 이미지 경로와 아이의 이름이 저장된다. 이 파일은 이후 일반 회원용 마이페이지에서 사용된다. mng\_version/adopt에는 [그림 27]과 같이 입양 문의를 받은 아이의 이름으로 파일이 생성된다. 파일 안에는 이미지 경로와 입양 문의를 한 사람의 정보가 저장된다. 이 파일은 이후 관리자용 마이페이지에서 사용된다.



[그림 26]



[그림 27]



#### \* children.js

```
let current_children = []; // 현재 사용자가 보고 있는 아이 데이터

// 입양 문의했을 경우 서버에 데이터 전송
function adopt_inquiry(){
    if($('#adopt_name').val() == "" || $('#adopt_tel').val() == "" || $('#address').val() == "" ||
    $('#job').val() == "" || $("input:checkbox[name=family]:checked").length == 0){
        alert("정보를 정확히 기입하세요.");
    }else{
        let family = [];
        $("input[name=family]:checked").each(function() {
            family.push($(this).val());
        });
        let obj = new Object();
        obj.adopt_name = $('#adopt_name').val();
        obj.adopt_birth_year = find_selected('adopt_birth_year');
        obj.adopt_birth_month = find_selected('adopt_birth_month');
        obj.adopt_birth_day = String(find_selected('adopt_birth_day'));
        obj.adopt_tel = $('#adopt_tel').val();
        obj.address = $('#address').val();
        obj.job = $('#job').val();
        obj.family = family;
        obj.have_experience = find_checked("have_experience");
        obj.current_children = current_children;
        obj = JSON.stringify(obj);
        $.post("./php/add_adopt.php",
            {
                data: obj
            },
            function(data, status) {
                alert(data);
            });
    }
}
```

Learn more를 클릭하면 current\_children에 현재 사용자가 보고 있는 아이의 데이터가 저장된다. 모든 정보가 기입됐는지 확인하고 current\_children과 입양 문의한 사람의 정보를 add\_adopt.php 서버로 보낸다.

#### \* add\_adopt.php

```
3 // add_adopt.php
4 // 입양 문의 데이터를 파일에 저장 및 저장 여부를 클라이언트에게 전송
5
6 if ($_SERVER["REQUEST_METHOD"] == "POST") {
7     $adopt_info = $_POST['data'];
8     $adopt_info_dec = json_decode($adopt_info, true);
9     $adopt_name = $adopt_info_dec["adopt_name"];
10    $adopt_tel = $adopt_info_dec["adopt_tel"];
11    $adopt_birth_year = $adopt_info_dec["adopt_birth_year"];
12    $adopt_birth_month = $adopt_info_dec["adopt_birth_month"];
13    $adopt_birth_day = $adopt_info_dec["adopt_birth_day"];
14    $job = $adopt_info_dec["job"];
15    $family = $adopt_info_dec["family"];
16    $address = $adopt_info_dec["address"];
17    $have_experience = $adopt_info_dec["have_experience"];
18    $current_children = $adopt_info_dec["current_children"];
19 }
```

```

20 // 입양 문의 받은 아이의 이름을 파일명으로 데이터 저장
21 $file_path = "../data/mng_version/adopt/".$current_children[1].".json";
22 $file = fopen($file_path, "a+");
23 $adopt_birth = $adopt_birth_year."/". $adopt_birth_month."/". $adopt_birth_day;
24 $adopt = array("img"=>$current_children[0], "name"=>$adopt_name, "birth"=>$adopt_birth, "tel"=>$adopt_tel,
25 "job"=>$job, "address"=>$address, "family"=>$family, "have_experience"=>$have_experience);
26 fwrite($file, json_encode($adopt, JSON_UNESCAPED_UNICODE)."\n"); // encode 해서 파일에 작성
27 fclose($file);
28
29 session_start();
30 if(isset($_SESSION['user_id'])){ // 로그인한 상태면 아이디를 파일명으로 데이터 저장
31     $user_v_file = fopen("../data/user_version/adopt/".$_SESSION['user_id'].".json", "a+");
32 }else{ // 비로그인 상태이면 신청자 이름을 파일명으로 데이터 저장
33     $user_v_file = fopen("../data/user_version/adopt/".$adopt_name.".json", "a+");
34 }
35 $adopt_content = array("img"=>$current_children[0], "pet_name"=>$current_children[1]);
36 fwrite($user_v_file, json_encode($adopt_content, JSON_UNESCAPED_UNICODE)."\n");
37 fclose($user_v_file);
38
39 echo "신청되었습니다.";

```

mng\_version/adopt에는 입양 문의를 받은 아이의 이름을 파일명으로 데이터를 저장한다.

로그인 상태이면 아이디를 파일명으로 데이터를 저장한다. 비로그인 상태이면 문의할 때 작성한 이름을 파일명으로 데이터를 저장한다. 저장 완료하면 '신청되었습니다.'를 클라이언트에게 전송한다.

### ③ 결연 맺기

\* 결연 맺기 버튼을 클릭하면 [그림 28]과 같이 결연 맺기 레이어가 나타난다.

반려동물 테스트

**결연 맺기**

이름 :

휴대폰 번호 :  010부터 ' 하이픈 없이 입력

주소 :

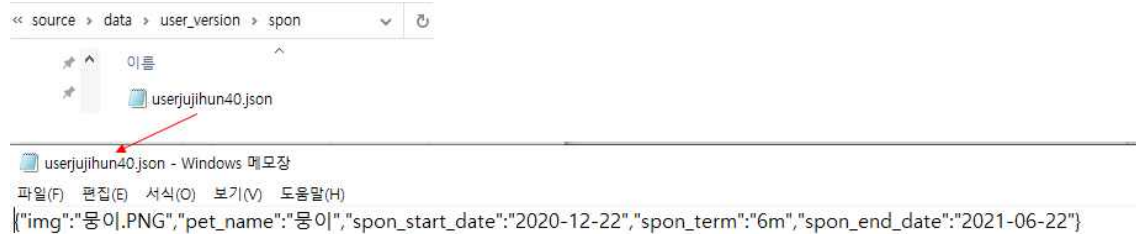
결연 기간 : 1개월 ▼

결연 맺기

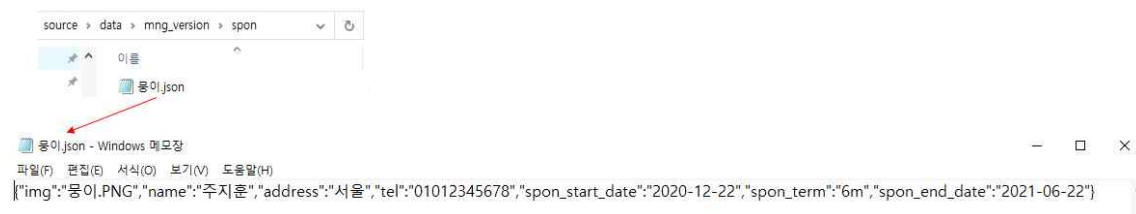
[그림 28]



\* 정보를 입력하고 확인 버튼을 클릭하면 user\_version/spon에는 [그림 29]와 같이 결연 맺기 후원을 한 회원의 아이디로 파일이 생성된다. 파일 안에는 이미지 경로, 아이의 이름, 결연 시작 날짜, 결연 기간, 결연 종료 날짜가 저장된다. 이 파일은 이후 일반 회원용 마이페이지에서 사용된다. mng\_version/spon에는 [그림 30]과 같이 후원을 받은 아이의 이름으로 파일이 생성된다. 파일 안에는 이미지 경로, 후원한 사람의 정보가 저장된다. 이 파일은 이후 관리자용 마이페이지에서 사용된다.



[그림 29]



[그림 30]

```
* children.js

613 // 결연 맺기 했을 경우 서버에 데이터 전송
614 function spon(){
615     if($('#spon_name').val() == "" || $('#spon_tel').val() == "" || $('#spon_address').val() == ""){
616         alert("정보를 정확히 기입하세요.");
617     }else{
618         var date = new Date();
619         var year = date.getFullYear();
620         var month = ("0" + (1 + date.getMonth())).slice(-2);
621         var day = ("0" + date.getDate()).slice(-2);
622         let obj = new Object();
623         obj.spon_name = $('#spon_name').val();
624         obj.spon_tel = $('#spon_tel').val();
625         obj.spon_address = $('#spon_address').val();
626         obj.spon_term = find_selected('spon_term');
627         obj.spon_date = year + "-" + month + "-" + day;
628         obj.current_children = current_children;
629         obj = JSON.stringify(obj);
630         $.post("./php/add_spon.php",
631             {
632                 data: obj
633             },
634             function(data, status) {
635                 alert(data);
636             });
637     }
638 }
```

모든 정보가 기입됐는지 확인하고 current\_children과 후원 시작 날짜, 후원한 사람의 정보를 add\_spon.php 서버로 보낸다.

\* add\_spon.php

```
3 // add_spon.php
4 // 결연 맺기 데이터를 파일에 저장 및 저장 여부를 클라이언트에게 전송
5
6 if ($_SERVER["REQUEST_METHOD"] == "POST") {
7     $spon_info          = $_POST['data'];
8     $spon_info_dec       = json_decode($spon_info, true);
9     $spon_name           = $spon_info_dec["spon_name"];
10    $spon_tel            = $spon_info_dec["spon_tel"];
11    $spon_address         = $spon_info_dec["spon_address"];
12    $spon_term            = $spon_info_dec["spon_term"];
13    $spon_date            = $spon_info_dec["spon_date"];
14    $current_children     = $spon_info_dec["current_children"];
15
16    $spon_end_date = 0;
17    // 후원 시작 날짜와 후원 기간으로 후원 종료 날짜 계산
18    if($spon_term == "1m"){
19        $spon_end_date = strtotime($spon_date.'+1 month');
20    }else if($spon_term == "6m"){
21        $spon_end_date = strtotime($spon_date.'+6 month');
22    }else if($spon_term == "1y"){
23        $spon_end_date = strtotime($spon_date.'+1 year');
24    }
25    $spon_end_date = date("Y-m-d", $spon_end_date);
26 }
```

클라이언트에게 받은 후원 시작 날짜와 후원 기간으로 후원 종료 날짜를 계산해서 파일에 저장한다. 파일 저장 동작 방식은 add\_adopt.php와 같다.

#### ④ 자원봉사 신청 - volunteer.html

- \* 자원봉사 신청 페이지의 달력에서 '신청하기' 버튼을 눌러 자원봉사를 신청할 수 있다.
- \* 오늘 이전의 날짜들은 빨간 버튼으로 표시되고 클릭하면 [그림 34]와 같이 '신청할 수 없는 날짜입니다' 알림창이 나타난다.
- \* 초록색 버튼을 클릭하면 [그림 35]와 같이 신청창이 나타난다. 만약 신청한 인원이 5명 이상이면 '인원이 다 찼습니다.' 알림창이 나타난다.

#### 예약 안내

- 원하시는 날짜의 예약하기 버튼을 클릭하고 예약해주세요.
- 행사가 없는 날의 제한 인수는 10명입니다.
- 당일 취소 및 무단 불참 시 추후 봉사 신청에 어려움이 생길 수 있습니다.
- 단체 봉사는 전화로 문의주세요.

2020년 12월						
일	월	화	수	목	금	토
		1 신청하기	2 신청하기	3 신청하기	4 신청하기	5 신청하기
6 신청하기	7 신청하기	8 신청하기	9 신청하기	10 신청하기	11 신청하기	12 신청하기
13 신청하기	14 신청하기	15 신청하기	16 신청하기	17 신청하기	18 신청하기	19 신청하기
20 신청하기	21 신청하기	22 신청하기	23 신청하기	24 신청하기	25 신청하기	26 신청하기
27 신청하기	28 신청하기	29 신청하기	30 신청하기	31 신청하기		

[그림 33]

#### 예약 안내

- 원하시는 날짜의 예약하기 버튼을 클릭하고 예약해주세요.
- 행사가 없는 날의 제한 인수는 10명입니다.
- 당일 취소 및 무단 불참 시 추후 봉사 신청에 어려움이 생길 수 있습니다.
- 단체 봉사는 전화로 문의주세요.

localhost 내용:  
신청할 수 없는 날짜입니다.

확인

2020년 12월						
일	월	화	수	목	금	토
		1 신청하기	2 신청하기	3 신청하기	4 신청하기	5 신청하기

[그림 34]

2020년 12월						
일	월	화	수	목	금	토
		1 신청하기	2 신청하기	3 신청하기	4 신청하기	5 신청하기
6 신청하기	7 신청하기	8 신청하기	9 신청하기	10 신청하기	11 신청하기	12 신청하기
13 신청하기	14 신청하기	15 신청하기	16 신청하기	17	18	19
20 신청하기	21 신청하기	22 신청하기	23 신청하기			
27 신청하기	28 신청하기	29 신청하기	30 신청하기			

2020/12/23

**자원봉사 신청**

이름

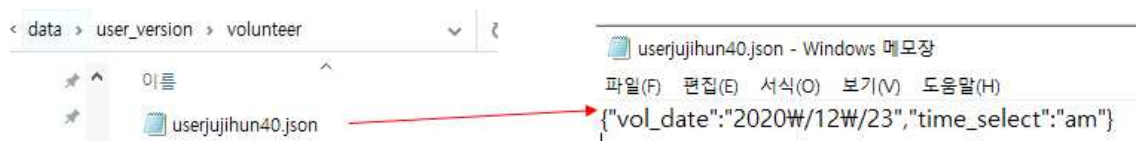
생년월일  년  월  일

휴대폰 번호

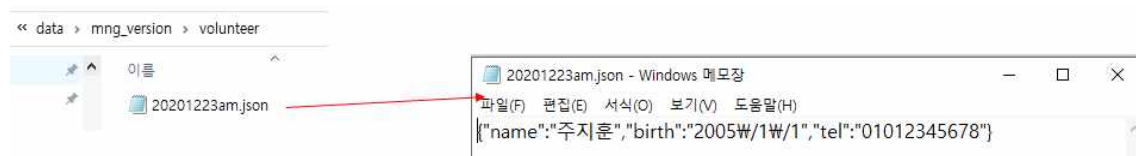
시간 ☒ 오전(9:00~12:00) ☐ 오후(14:00~17:00)

[그림 35]

\* 신청이 완료되면 user\_version/volunteer에는 [그림 36]과 같이 신청한 회원의 아이디로 파일이 생성된다. 파일 안에는 자원봉사 날짜와 시간이 저장된다. 이 파일은 이후 일반 회원용 마이페이지에서 사용된다. mng\_version/volunteer에는 [그림 37]과 같이 봉사활동 날짜와 시간을 파일명으로 파일이 생성된다. 파일 안에는 해당 날짜와 시간에 자원봉사를 신청한 사람의 정보가 저장된다. 이 파일은 이후 관리자용 마이페이지에서 사용된다.



[그림 36]



[그림 37]

\* volunteer.js

```
109 // 자원봉사 신청 레이어에서 확인 버튼 눌렀을 경우
110 ~ function add_volunteer(year, month, day){
111 ~   if($('#name').val() == "" || $('#vol_tel').val() == ""){
112     alert("정보를 정확히 기입하세요.");
113 ~   }else{ // 서버에 전송해서 데이터 받아옴
114     let obj = new Object();
115     obj.vol_name = $('#vol_name').val();
116     obj.vol_tel = $('#vol_tel').val();
117     obj.vol_birth_year = find_selected('vol_birth_year');
118     obj.vol_birth_month = find_selected('vol_birth_month');
119     obj.vol_birth_day = String(find_selected('vol_birth_day'));
120     obj.time_select = find_checked("time_select");
121     obj.year = year;
122     obj.month = month;
123     obj.day = day;
124     obj = JSON.stringify(obj);
125 ~   $.post("./php/volunteer.php",
126 ~     {
127       data: obj
128     },
129 ~     function(data, status) {
130       alert(data);
131     });
132   }
133 }
```

자원봉사 신청 레이어에서 확인 버튼을 누르면 모든 정보가 기입되었는지 확인하고 volunteer.php 서버에 전송해서 데이터를 받는다.

\* volunteer.php

```
1 <?php
2
3 // volunteer.php
4 // 자원봉사자 정보를 파일에 저장
5
6 if ($_SERVER["REQUEST_METHOD"] == "POST") {
7   $vol_info      = $_POST['data'];
8   $vol_info_dec  = json_decode($vol_info, true);
9   $vol_name      = $vol_info_dec["vol_name"];
10  $vol_tel       = $vol_info_dec["vol_tel"];
11  $vol_birth_year = $vol_info_dec["vol_birth_year"];
12  $vol_birth_month = $vol_info_dec["vol_birth_month"];
13  $vol_birth_day  = $vol_info_dec["vol_birth_day"];
14  $time_select    = $vol_info_dec["time_select"];
15  $year          = $vol_info_dec["year"];
16  $month         = $vol_info_dec["month"];
17  $day           = $vol_info_dec["day"];
```

클라이언트에게 받은 자원봉사자 정보를 decode해서 변수에 저장한다.

```

19 // 해당 날짜에 저장된 인원이 5명 이상이면 저장하지 않음. 파일이 없거나 인원이 5명 미만이면 저장
20 $check = true;
21 $file_path = "../data/mng_version/volunteer/" . $year . $month . $day . $time_select . ".json";
22 if(file_exists($file_path)){
23     $file = fopen($file_path, "r");
24     if(count_vol_num($file) >= 5){
25         $check = false;
26     }
27     fclose($file);
28 }
29 if($check){ // 파일이 없거나 인원이 5명 미만일 경우
30     $file = fopen($file_path, "a+");
31     $vol_birth = $vol_birth_year . "/" . $vol_birth_month . "/" . $vol_birth_day;
32     $vol = array("name" => $vol_name, "birth" => $vol_birth, "tel" => $vol_tel);
33     fwrite($file, json_encode($vol, JSON_UNESCAPED_UNICODE) . "\n"); // encode 해서 파일에 작성
34     fclose($file);
35     session_start();
36     if(isset($_SESSION['user_id'])){ // 로그인 상태면 아이디를 파일명으로 저장, 비로그인 상태이면 신청자 이름으로 파일명 저장
37         $user_v_file = fopen("../data/user_version/volunteer/" . $_SESSION['user_id'] . ".json", "a+");
38     }else{
39         $user_v_file = fopen("../data/user_version/volunteer/" . $vol_name . ".json", "a+");
40     }
41     $vol_date = $year . "/" . $month . "/" . $day;
42     $user_vol_content = array("vol_date" => $vol_date, "time_select" => $time_select);
43     fwrite($user_v_file, json_encode($user_vol_content, JSON_UNESCAPED_UNICODE) . "\n");
44     fclose($user_v_file);
45     echo "신청되었습니다.";
46 }else{ // 인원이 5명 이상일 경우
47     echo "인원이 다 찼습니다.";
48 }
49 }

```

mng\_version/volunteer에는 날짜와 시간을 파일명으로 데이터가 저장되는데 파일이 없거나 신청한 인원이 5명 미만이면 파일을 생성한 후 데이터를 저장한다. 5명 이상이면 저장하지 않는다. user\_version/volunteer에는 로그인 상태일 경우 아이디를 파일명으로 데이터를 저장하고 비로그인 상태일 경우 신청자 이름을 파일명으로 데이터를 저장한다.

```

51 function count_vol_num($file){ // 파일에 저장된 자원봉사자 인원 카운팅해서 반환
52     $count = 0;
53     while (!feof($file)) {
54         $line = json_decode(fgets($file), true);
55         if ($line != "") {
56             $count = $count + 1;
57         }
58     }
59     return $count;
60 }

```

count\_vol\_num 함수는 파일에 저장된 신청 인원을 카운팅해서 반환한다.

##### ⑤ 마이페이지 - myPage.html

\* 마이페이지 [그림 42]는 ‘입양 문의 중인 아이’, ‘후원 중인 아이’, ‘자원봉사 신청 내역’ 세 영역으로 나뉜다.


\* 로그아웃 버튼을 클릭하면 로그아웃할 수 있다.




마이 페이지

[로그아웃]

입양 문의 중인 아이

이미지	이름
	땡이

후원 중인 아이

이미지	이름	후원시작일	후원종료일	후원 기간	디데이
	뭉이	2020-12-22	2021-06-22	6개월	D-181

자원봉사 신청 내역

날짜	시간
2020/12/23	오전(9:00~12:00)

[그림 42]

\* myPage.js

```

2 // Logout 버튼 클릭
3 $("#logout_btn").click(function() {
4     $.post("php/logout.php");
5     history.back();
6 });

```

로그아웃 버튼을 클릭하면 logout.php 서버 요청한다.

\* logout.php

```

1 <?php
2
3 // Logout.php
4 // 세션 삭제해서 로그아웃
5
6     session_start();
7     session_destroy(); // 세션 삭제
8 ?>

```

세션을 삭제해서 로그아웃 한다.

\* spon-ing.php

```
46 }else{ // 일반회원 로그인시 개별 후원 내역을 전송
47     $user_id = $_SESSION['user_id'];
48     $dir = "../data/user_version/spon";
49     $handle = opendir($dir);
50     $files = array();
51     while (false != ($filename = readdir($handle))) {
52         if ($filename == "." || $filename == "..") {
53             continue;
54         }
55         if (is_file($dir . "/" . $filename)) {
56             // 회원 아이디로 저장된 파일명만 $files에 모음
57             $id_in_filename = explode('.', $filename);
58             if ($id_in_filename[0] == $user_id) {
59                 $files[] = $filename;
60             }
61         }
62     }
63     closedir($handle);
64
65     if (count($files) != 0) {
66         $all_list = array();
67         foreach ($files as $f) {
68             $f_open = fopen($dir . "/" . $f, 'r');
69             while (!feof($f_open)) {
70                 $line = json_decode(fgets($f_open));
71                 if ($line != "") {
72                     array_push($all_list, $line); // 파일 내용을 모두 $all_list에 저장
73                 }
74             }
75             fclose($f_open);
76         }
77         if (count($all_list) != 0) {
78             $send_data = array();
79             $send_data['type'] = $_SESSION['type'];
80             $send_data['all_list'] = $all_list;
81             echo json_encode($send_data); // 로그인 타입과 자원봉사 신청 내역을 배열로 묶어 클라이언트에게 전송
82         }
83     }
84 }
```

\$\_SESSION['type']에는 관리자 로그인이면 mng가 저장되어 있고 일반 회원 로그인이면 user가 저장되어 있다. \$\_SESSION['type']으로 관리자 로그인인지, 일반 회원 로그인인지 검사한다. 관리자 로그인일 경우는 이후 관리자 로그인 시나리오에서 설명하겠다.

\$\_SESSION['type']가 user일 경우 일반 회원 로그인이기 때문에 개별 후원 내역을 클라이언트에게 전송한다. 우선 user\_version/spon에서 회원 아이디로 저장된 파일명들을 모아 하나씩 파일을 연다. 파일 내용(후원 내역)들을 저장한 배열과 로그인 타입을 다시 배열로 묶어 클라이언트에게 전송한다.

\* myPage.js



```

221 // 서버로부터 후원(결연 맺기) 내역 불러오기
222 $.post("php/spon-ing.php",
223     function(data) {
224         var spon_tab = document.getElementById('spon_tab');
225         if (data == '') { // 서버로부터 받은 데이터가 없을 경우
226             var tr = document.createElement('tr');
227             var td1 = document.createElement('td');
228             td1.innerText = "후원 내역이 없습니다";
229             tr.appendChild(td1);
230             spon_tab.appendChild(tr);
231         } else { // 서버로부터 받은 데이터가 있을 경우
232             data = JSON.parse(data);
233             // 일반 회원 마이페이지 후원 내역 : 이미지, 이름, 후원시작일, 후원종료일, 후원기간, 디데이
234             if (data['type'] == 'user') {
235                 var tr1 = document.createElement('tr');
236                 var img_th = document.createElement('th');
237                 var petname_th = document.createElement('th');
238                 var start_date_th = document.createElement('th');
239                 var end_date_th = document.createElement('th');
240                 var term_th = document.createElement('th');
241                 var dday_th = document.createElement('th');
242
243                 img_th.innerText = "이미지";
244                 petname_th.innerText = "이름";
245                 start_date_th.innerText = "후원시작일";
246                 end_date_th.innerText = "후원종료일";
247                 term_th.innerText = "후원 기간";
248                 dday_th.innerText = "디데이";
249
250                 tr1.appendChild(img_th);
251                 tr1.appendChild(petname_th);
252                 tr1.appendChild(start_date_th);
253                 tr1.appendChild(end_date_th);
254                 tr1.appendChild(term_th);
255                 tr1.appendChild(dday_th);
256                 spon_tab.appendChild(tr1);

```

마이페이지에 들어오면 spon-ing.php 서버로부터 후원 내역들과 로그인 타입을 받는다. 로그인 타입이 user일 경우 후원 내역으로 일반 회원 마이페이지를 구성한다. 일반 회원용 마이페이지의 후원 내역에는 이미지, 이름, 후원시작일, 후원종료일, 후원기간, 디데이가 들어간다.

```

258 ~      for (var i = 0; i < data['all_list'].length; i++) {
259          var tr2 = document.createElement('tr');
260          var img_td = document.createElement('td');
261          var petname_td = document.createElement('td');
262          var start_date_td = document.createElement('td');
263          var end_date_td = document.createElement('td');
264          var term_td = document.createElement('td');
265          var dday_td = document.createElement('td');
266          var img1 = document.createElement('img');
267
268          img1.setAttribute('src', data['all_list'][i]['img']);
269          petname_td.innerText = data['all_list'][i]['pet_name'];
270          start_date_td.innerText = data['all_list'][i]['spon_start_date'];
271          end_date_td.innerText = data['all_list'][i]['spon_end_date'];
272
273          // 결연 기간 한글로 변환
274 ~      switch (data['all_list'][i]['spon_term']) {
275 ~          case '1m':
276              term_td.innerText = "1개월";
277              break;
278 ~          case '6m':
279              term_td.innerText = "6개월";
280              break;
281 ~          case '1y':
282              term_td.innerText = "1년";
283              break;
284      }

```

서버로부터 받은 결연 기간은 '1m', '6m', '1y'와 같이 저장되어있으므로 알아보기 쉽게 한글로 변환해서 마이페이지에 띄운다.

```

286          // 디데이 표시
287          var today = new Date();
288          let strArr = data['all_list'][i]['spon_end_date'].split('-');
289          let end_date = new Date(strArr[0], strArr[1] - 1, strArr[2]);
290          var gap = end_date.getTime() - today.getTime();
291          gap = Math.floor(gap / (1000 * 60 * 60 * 24));
292 ~      if (gap < 0) {
293          dday_td.innerText = "종료";
294 ~      } else {
295          dday_td.innerText = "D-" + gap;
296      }
297
298 ~      img_td.appendChild(img1);
299 ~      tr2.appendChild(img_td);
300 ~      tr2.appendChild(petname_td);
301 ~      tr2.appendChild(start_date_td);
302 ~      tr2.appendChild(end_date_td);
303 ~      tr2.appendChild(term_td);
304 ~      tr2.appendChild(dday_td);
305 ~      spon_tab.appendChild(tr2);
306 ~      }

```

오늘 날짜와 후원종료일 간의 차이를 구해서 디데이를 표시한다. 만약 종료일을 넘어서면 '종료'라고 띄운다.

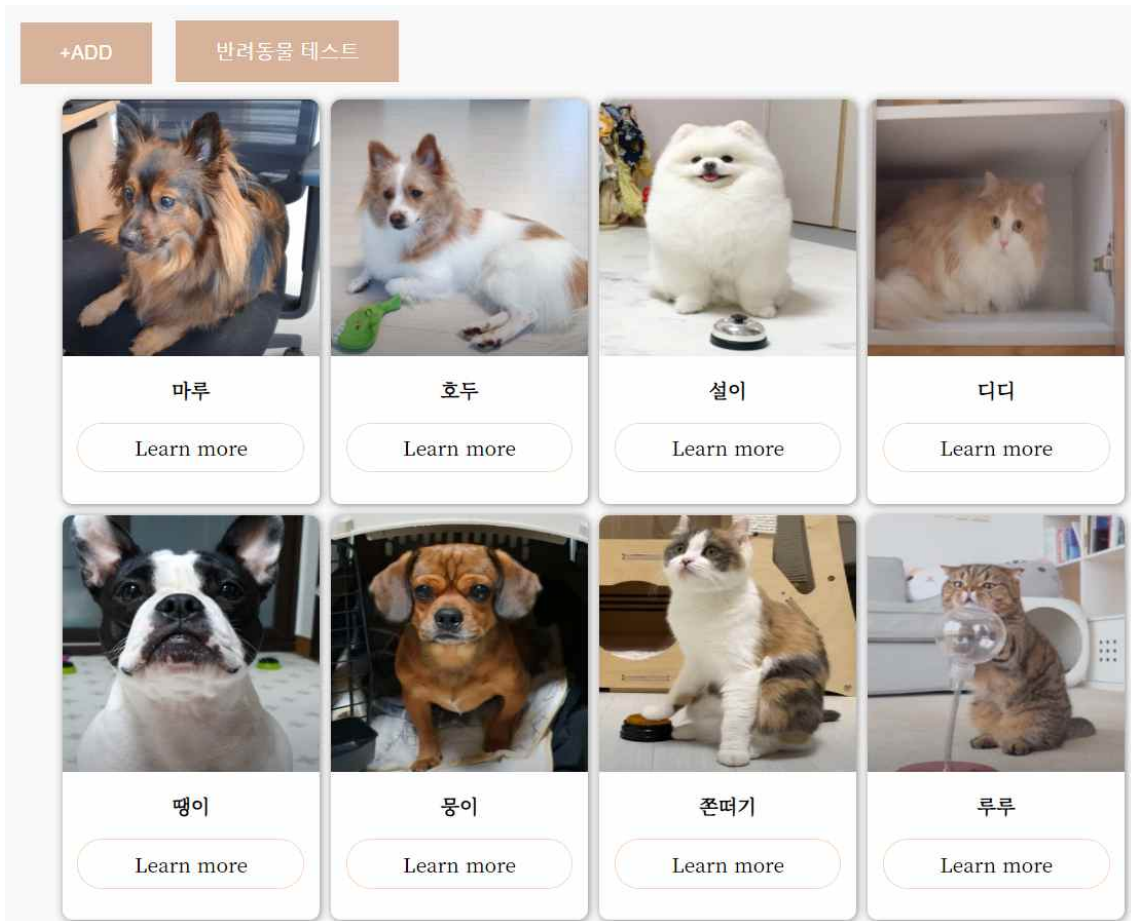
입양 문의 내역, 자원 봉사 신청 내역을 서버로부터 받아오는 과정도 위와 비슷하기 때문에 설명은 생략한다.

(5) 관리자 시나리오

① 새로운 아이 등록 - children.html


\* 관리자로 로그인한 경우 '센터의 아이들' 페이지에 [그림 49]와 같이 '+ADD' 버튼이 나타난다.

\* '+ADD' 버튼을 클릭하면 [그림 50]과 같이 새로운 아이 등록 레이어가 나타난다.




[그림 49]

+ADD



마루

Learn more



땡이

Learn more

새로운 아이 등록

이미지 파일

파일 선택 선택된 파일 없음

이름

성별

☐ 남 ☐ 여

분류

☐ 개 ☐ 고양이

입양 상태

☐ 입양 ☒ 미입양

입소 날짜

연도-월-일

성격

☐ 분리불안 ☐ 애정결핍 ☐ 온순함 ☐ 사나움 ☐ 조용함  
☐ 활발함

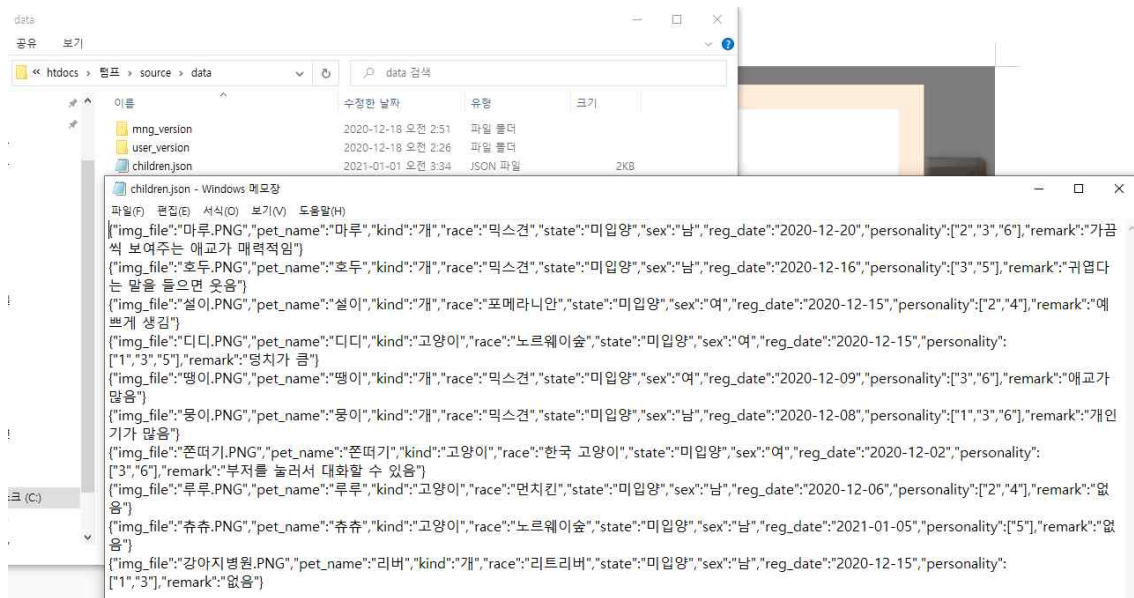
특이사항

등록

취소

[그림 50]

\* 정보를 입력하고 등록 버튼을 누르면 children.json 파일에 [그림 51]과 같이 정보가 저장된다.



[그림 51]

#### \* children.json

```

221 // 등록 데이터를 JSON 인코딩해서 서버로 전송하고 결과를 받아서 카드 생성
222 function add_file(img_file, pet_name, sex_val, kind_val, race, state_val, reg_date, personality, remark) {
223     let obj = new Object();
224     obj.img_file = img_file;
225     obj.pet_name = pet_name;
226     obj.kind_val = kind_val;
227     obj.sex_val = sex_val;
228     obj.race = race;
229     obj.state_val = state_val;
230     obj.reg_date = reg_date;
231     obj.personality = personality;
232     obj.remark = remark;
233     obj = JSON.stringify(obj);
234     $.post("./php/children.php", // POST 방식으로 데이터 전송
235     {
236         data: obj
237     },
238     function(data, status) { // 콜백 함수. 전송받은 데이터와 전송 성공 여부를 보여줌.
239         let wrap_div = document.getElementsByClassName('wrap')[0];
240         if (document.getElementById('no_content')) {
241             wrap_div.removeChild(document.getElementById('no_content'));
242         }
243         data = JSON.parse(data);
244         // 서버로부터 받은 데이터를 디코딩해서 카드 생성
245         // 이때 받는 데이터는 모든 아이가 아니라 등록된 아이 하나에 대한 데이터
246         make_card(data["img_file"], data["pet_name"], data["sex"], data["kind"], data["race"],
247         data["state"], data["reg_date"], data["personality"], data["remark"]);
248     });
249 }
250 }

```

등록 버튼을 누르면 정보가 제대로 기입되었는지 확인하고 children.php 서버로 데이터를 보낸다. 이때 서버로부터 받는 데이터는 모든 아이가 아니라 아이 하나에 대한 데이터이다. 데이터를 받아서 make\_card 함수를 이용해서 카드를 생성한다.

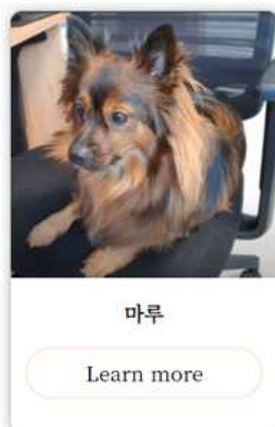


```

307 // 카드 생성
308 v function make_card(img_file, pet_name, sex, kind, race, state, reg_date, personality, remark) {
309     let card_list = document.getElementById('card_list');
310     let add_li = document.createElement('li');
311     let item = document.createElement('div');
312     let item_img = document.createElement('div');
313     let img = document.createElement('img');
314     let p_tag = document.createElement('p');
315     let learn_more = document.createElement('a');
316
317     item.setAttribute('class', 'item');
318     item_img.setAttribute('class', 'item_img');
319
320     img.setAttribute('src', img_file);
321     img.setAttribute('alt', img_file);
322     item_img.appendChild(img);
323
324     p_tag.innerHTML = pet_name;
325     p_tag.setAttribute('id', pet_name);
326
327     learn_more.setAttribute('href', '#none');
328     learn_more.innerHTML = "Learn more";
329
330     // learn_more 버튼 클릭하면 디테일 카드 생성
331 v learn_more.addEventListener("click", function() {
332     make_detail_card(img_file, pet_name, sex, kind, race, state, reg_date, personality, remark);
333     current_children = [img_file, pet_name];
334 });
335
336 item.appendChild(item_img);
337 item.appendChild(p_tag);
338 item.appendChild(learn_more);
339 add_li.appendChild(item);
340 card_list.appendChild(add_li);
341 }

```

make\_card 함수는 아래와 같은 카드를 만드는 함수이다. 'Learn more' 버튼을 클릭하면 디테일 카드를 생성하는 make\_detail\_card 함수를 호출한다. 입양 문의, 결연 맺기를 위해 current\_children에 현재 보고 있는 아이의 정보를 저장한다.



```

343 // 디테일 카드 생성
344 <v> function make_detail_card(img_file, pet_name, sex, kind, race, state, reg_date, personality, remark) {
345 // 디테일 카드 나가기 버튼과 디테일 카드 레이어 보여줄
346 show_popup("eixt_div");
347 show_popup("detail_card_div");
348
349 // 기존 디테일 카드 레이어 비우기
350 $("#detail_card_div").empty();
351
352 let detail_card_div = document.getElementById("detail_card_div");
353 let table = document.createElement('table');
354 let tbody = document.createElement('tbody');
355 let tr = document.createElement('tr');
356 let img_td = document.createElement('td');
357
358 let img = document.createElement('img'); // 디테일 카드 이미지
359 img.setAttribute('src', img_file);
360 img.setAttribute('alt', img_file);
361 img_td.appendChild(img);

```

```

363 let detail_content = document.createElement('td'); // 디테일 카드 본문
364 let card_title = document.createElement('span');
365 card_title.setAttribute('class', 'card_title');
366 card_title.innerHTML = "[" + kind + "]" + pet_name;
367 let ul = document.createElement("ul");
368 let sex_li = document.createElement("li");
369 sex_li.innerHTML = "성별 : " + sex;
370 let race_li = document.createElement("li");
371 race_li.innerHTML = "품종 : " + race;
372 let state_li = document.createElement("li");
373 state_li.innerHTML = "입양 상태 : " + state;
374 let reg_date_li = document.createElement("li");
375 reg_date_li.innerHTML = "입소 날짜 : " + reg_date;
376 let personality_li = document.createElement("li");
377 personality_li.innerHTML = "성격 : " + personality_to_lang(personality);
378 let remark_li = document.createElement("li");
379 remark_li.innerHTML = "특이사항 : " + remark;
380
381 ul.appendChild(sex_li);
382 ul.appendChild(race_li);
383 ul.appendChild(state_li);
384 ul.appendChild(reg_date_li);
385 ul.appendChild(personality_li);
386 ul.appendChild(remark_li);
387
388 let btns_div = document.createElement('div'); // 입양 문의 버튼
389 btns_div.setAttribute('class', 'btn_group');
390 let adopt_btn = document.createElement('button');
391 adopt_btn.innerHTML = "입양 문의";
392 <v> adopt_btn.addEventListener("click", function() {
393 show_popup('adopt_inquiry_div');
394 });
395

```

make\_detail\_card 함수는 아래와 같은 디테일 카드를 생성하는 함수이다. table 안에 하나의 tr이 있고 tr 안에 이미지 td 하나, 본문 td 하나로 구성되어있다.



\* children.php

```

1 <?php
2
3 // children.php
4 // 새로운 아이 등록시 children.json 파일에 저장
5
6 function test_input($data) {
7     $data = trim($data);
8     $data = stripslashes($data);
9     $data = htmlspecialchars($data);
10    return $data;
11 }
12
13 if ($_SERVER["REQUEST_METHOD"] == "POST") {
14     $obj = $_POST["data"];
15     $obj_dec = json_decode($obj, true);
16     // validation
17     $img_file = test_input($obj_dec["img_file"]);
18     $pet_name = test_input($obj_dec["pet_name"]);
19     $sex_val = test_input($obj_dec["sex_val"]);
20     $kind_val = test_input($obj_dec["kind_val"]);
21     $race = test_input($obj_dec["race"]);
22     $state_val = test_input($obj_dec["state_val"]);
23     $reg_date = test_input($obj_dec["reg_date"]);
24     $personality = $obj_dec["personality"];
25     $remark = test_input($obj_dec["remark"]);
26
27     // children.json 파일에 데이터 추가
28     $file = fopen("../data/children.json", "a+");
29     $input = array("img_file"=>$img_file, "pet_name"=>$pet_name, "kind"=>$kind_val, "race"=>$race,
30         "state"=>$state_val, "sex"=>$sex_val, "reg_date"=>$reg_date, "personality"=>$personality, "remark"=>$remark);
31     $json_input = json_encode($input, JSON_UNESCAPED_UNICODE); // 한글 깨지지 않도록 옵션 추가
32     fwrite($file, $json_input . "\n"); // encode 해서 파일에 작성
33     fclose($file);
34     echo $json_input; // 등록된 아이에 대한 데이터를 클라이언트에게 넘겨줌
35 }
36
37 ?>

```

클라이언트에게 받은 정보를 children.json 파일에 저장하고 등록된 아이에 대한 데이터를 클라이언트에게 전송한다.

## ② 마이페이지 - myPage\_mng.html

\* 관리자용 마이페이지에서는 [그림 59]와 같이 모든 회원, 비회원의 입양 문의 명단, 후원 명단, 자원봉사 신청 명단을 확인할 수 있다.





[센터소개](#)[센터의 아이들](#)[입양](#)[후원](#)[자원봉사](#)[마이페이지](#)



## 마이 페이지

[\[로그아웃\]](#)

### 입양 문의 명단

멍이	이름	생년월일	연락처	직업	주소	가족구성원	반려동물 양육 경험
	주지훈	2001/01/01	01012345678	배우	서울	없음	무
호두	이름	생년월일	연락처	직업	주소	가족구성원	반려동물 양육 경험
	조은빈	2000/01/01	01095513735	학생	경상북도 구미시	부/모, 형제	유

### 후원 명단

몽이	이름	주소	연락처	후원시작일	후원종료일	후원 기간	디데이
	주지훈	서울	01012345678	2020-12-22	2021-06-22	6개월	D-181
설이	이름	주소	연락처	후원시작일	후원종료일	후원 기간	디데이
	조은빈	경상북도 구미시	01095513735	2021-01-01	2021-07-01	6개월	D-190

### 자원봉사 신청 명단

20201223am	이름	생년월일	연락처
	주지훈	2005/1/1	01012345678

[그림 59]

(6) 센터 소개 - introduce.html

\* 센터 소개를 확인한다.

 LiveUs

센터소개   센터의 아이들   입양   후원   자원봉사   마이페이지



LiveUs

LiveUs를 소개합니다

LiveUs는 안락사 없는 유기동물 보호소입니다. 보호소로 들어오는 아이들은 꼼꼼한 심사 절차를 거쳐 좋은 가족에게 입양을 가거나 보호소에서 생이 다할 때까지 케어를 받으며 생활하게 됩니다.



**철창 없는 따뜻한 보금자리**

LiveUs는 유기, 피양되는 아이들을 보호하고 따뜻한 보금자리가 되어주고 있습니다. 절대 좁은 철창에 가두지 않으며 충분한 생활 공간을 제공하고 프라이빗한 공간에서 생활할 수 있도록 합니다.



**다채로운 프로그램**

LiveUs는 아이들의 신체적, 정신적 건강을 위한 '우디다 운동회', '공공 현장체험', '목욕의 날' 등 다채로운 프로그램을 진행합니다. LiveUs는 아이들이 즐거운 생활을 할 수 있도록 최선을 다합니다.



**철저한 헬스케어**

LiveUs는 센터의 전체 아이들을 대상으로 정기적인 건강검진을 진행합니다. 연령이 많거나 병을 가진 아이들에 대해서는 개별적인 케어를 진행합니다.

유기동물보호소 LiveUs  
Company Name : 펍바니유니버스  
사업자등록번호 : 000-00-00000  
대표 : 조은빈

## (7) 입양 - adopt.html

\* 유기견 입양 홍보용 동화와 입양 절차 동영상 및 공지를 확인한다.

LiveUs

센터소개

센터의 아이들

입양

후원

자원봉사

마이페이지

입양

아이들의 가족이 되어주세요.

이벤트에서 만날까지  
- 유기견 카라 이야기 -

ep12. 유기동물 입양 절차의 모든 것

입양 자격

책임입양이 가능한 만 19세 이상 신원 확인 가능한 성인

경제적 능력이 있고 반려동물을 사랑하는 가정

입양 후 1년간 6회 이내 아이 생활 사진을 보내주실 수 있는 분

입양 절차

1. 센터의 아이들 확인 후 센터 연락처로 문의

2. 센터 방문

입양하려는 아이와 미팅

3. 책임분양계약 및 입양

상담 및 분양계약서 작성

책임비 안내

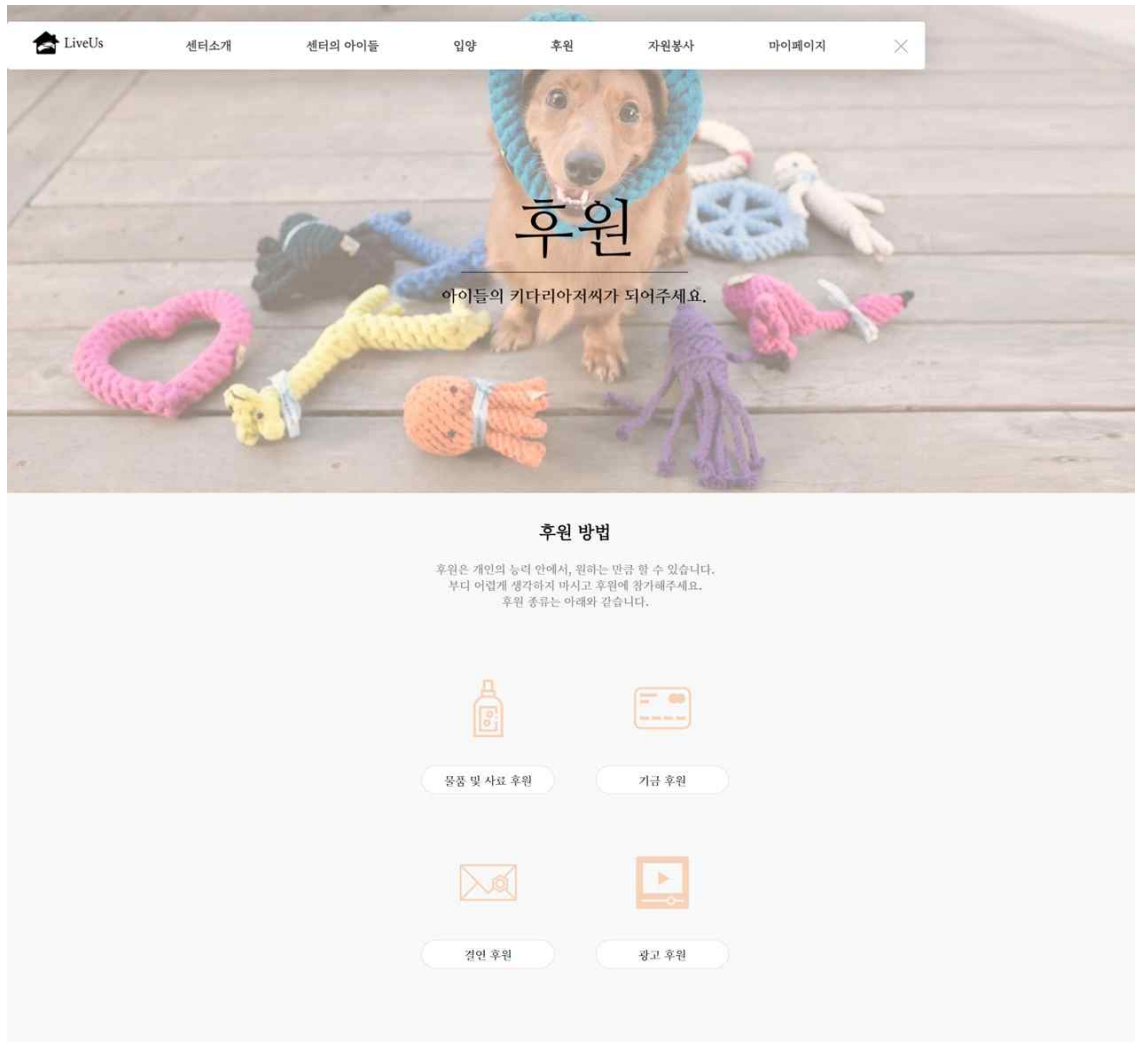
책임비는 수의 목적 재분양, 재활기비용, 식용도용, 재경능력 있는 가정을 필터링하기 위한 최소한의 장치입니다.

아이에 따라 무료분양 또는 소정의 책임비가 부여됩니다.

유기동물보호소 LiveUs  
Company Name : 비너컴퍼니  
사업자등록번호 : 000-00-00000  
대표 : 조은빈

(8) 후원

- \* 후원 방법을 자세히 설명하고 광고 후원에 참여할 수 있다.



---

## 물품 및 사료 후원

생활에 필요한 물품들과 사료를 후원 받습니다.

- ✓ 식기
- ✓ 화장실
- ✓ 동물 전용 삼투
- ✓ 장난감
- ✓ 수건
- ✓ 담요
- ✓ 배변패드
- ✓ 배변봉투
- ✓ 옷
- ✓ 눈, 귀 청소약
- ✓ 고양이 모래
- ✓ 고양이, 강아지 견인망 사료
- ✓ 고양이, 강아지 간식

후원하실 물품들을 아래 주소로 보내주세요.

주소 : 대전광역시 OO구 OO동 LiveUs

---

## 기금 후원

운영 및 치료에 필요한 기금을 후원 받습니다.

아래 계좌로 송금해주시고 품을 작성해주세요.  
기금 후원에 참가해주시는 분들께 감사의 편지와 함께 소정의 상품을 보내드립니다.

계좌 : 농협 000-0000-0000-00 조운빈

---

### 결연 후원

센터의 아이와 결연을 맺어 정기적으로 후원합니다.

센터의 아이들 중에 한 아이에게 마음에 거지만 입양하기는 어려운 상황이라면 결연을 맺어 정기적으로 후원하실 수 있습니다. 후원 기간은 3개월 단위로 이루어지며 1개월에 3만원으로 아이의 생활에 도움을 줄 수 있습니다. 결연 후원에 참가해주시는 분들께서는 감사의 편지와 함께 후원하시는 아이의 사진을 정기적으로 받아보실 수 있습니다. 자세한 문의는 센터로 연락 주시길 바랍니다.

연락처 : 123-4567-8910

결연 맺으러 가기

### 광고 후원

짧은 광고를 시청하면 아이들에게 후원됩니다.

아래에 게시된 짧은 광고를 시청하시면 소정의 수익금이 아이들에게 돌아옵니다. 눈 깜짝할 새에 지나가는 30초로 아이들의 삶이 바뀔까 생각하실 수 있습니다. 물론 한 명의 짧은 광고 시청만으로 아이들에게 가족이 생기고, 커다란 집이 생기지는 않습니다. 하지만 한 명의 참여가 모이고 모여 아이들에게 따뜻한 잠자리가 되고, 밥 한 끼가 됩니다.



## 4. 설계 1 보고서와의 다른 점

### (1) 구현하지 못한 부분



\* 설계 1 보고서 중 관리자로 로그인 하는 경우, 디테일 카드에 수정 버튼을 띄워서 수정할 수 있는 팝업창이 나타나도록 설계하였다. 설계 1 보고서 피드백을 받고 다른 질적인 요소에 집중하기 위해 구현하지 않았다.

(2) 추가로 구현한 부분

\* **반려동물 테스트** : 자신의 상황에 맞는 반려동물을 찾아준다. 상황을 묻는 질문에 O/X로 답하면 센터 내에서 그에 맞는 성격의 반려동물을 검색한다. 이름, 성격 검색보다 더 기능적이고 재미 요소를 추가하였다.

\* **입양 홍보 동화책** : 유기견이 입양되는 스토리를 동화책 형식으로 보여 유기견 입양 홍보 효과를 도모하고 시각적 재미 요소를 부여하였다.

\* **마이페이지** : 관리자용 마이페이지에서 **회원, 비회원**들의 입양 문의 내역, 결연 맺기 후원 내역, 자원봉사자 신청 내역들을 확인할 수 있다. 일반 회원용 마이페이지에서 **자신**의 입양 문의 내역, 결연 맺기 후원 내역, 자원봉사자 신청 내역들을 확인할 수 있다.

## 5. 노력 및 어려웠던 점

기능 위주의 웹 페이지를 설계했었는데 시각적 재미 요소가 있으면 좋겠다는 피드백을 받았다. 현재 레이아웃을 크게 바꾸지 않으면서 재미 요소를 추가할 방향을 잡는 게 어려웠다. 결과적으로 재미 요소를 많이 넣지 못한 점이 아쉽다.

너무 많은 기능을 계획했기 때문에 구현할 양이 많아진 점이 가장 힘들었다. 학기 초반부터 꾸준히 구현해왔지만 학기 초반에는 기능적 부분들을 구현할 능력이 부족해 레이아웃만 만들었다. 이후 자바스크립트와 제이쿼리를 배우면서 기능적인 코드를 작성하기 시작했는데 시간이 날 때마다 코드를 작성해도 시간이 모자랐다. 하지만 꾸준히 자료를 찾아보고 코드를 작성함으로써 웹 페이지 제작 능력도 향상시키고, 웹 페이지 제작에 재미도 붙여서 다른 웹 페이지도 만들어보고 싶다고 다짐하는 계기가 되었다.