

Patriots!

Location-Based WebApplication

팀 장 | 안준현

팀 원 | 김창기

| 김현우

| 박종호

| 이은채

| 성열원

Index

01 Intro	03
02 요구분석	05
03 기안서	09
04 배경지식	11
05 Database 모델링 & 테이블 명세서	20
06 StoryBorad	35
07 발표자료	76
08 기술문서	94
09 파일구조도	148
10 맺음말	157

Intro

Location-Based WebApplication

안녕하세요. TEAM_PETRIOTS입니다.

PETRIOTS는 PET(반려동물)과 PATRIOTS(애국자들)을 합성해서 만든 단어입니다.

펫에 대한 사랑을 가지고 있는 사람들이란 뜻이죠.

이는 PETRIOTS을 이용할 성실한 펫시터들을 뜻하기도,
PETRIOTS를 찾아주실 여러 반려인들을 뜻하기도 합니다.

물론, 반려동물들과 반려인들이 PETRIOTS로 인해
조금이라도 행복해지길 바라는 TEAM_PETRIOTS를 뜻하기도 합니다.

PETRIOTS를 구상하게 된 계기는
반려동물을 키우고 싶은 팀원의 바람에서 시작됐습니다.

자신이 반려인이 될 자격이 있는 사람인지,
반려인이 되더라도 직장생활 때문에
반려동물을 잘 돌 볼 수 있을지에 대한 우려로
반려동물을 기를 수 없는 팀원을 보고
사람들이 펫시팅을 간편하게 할 수 있는

또한,

반려동물을 기르고 싶은 사람은
자신이 반려동물을 돌보는 것을 좋아하는지
알아볼 수 있게끔 누구나 펫시터를 할 수 있는
웹 어플리케이션을 기획했습니다.

PETRIOTS는 가까운 거리의 펫시터를 조회할 수 있고
예약까지 할 수 있는 위치 기반 Web Application입니다.

요구분석

Location-Based WebApplication

● 관리자측 요구 분석

1. 시스템 구성 요구분석

- Windows 플랫폼의 Spring MVC 기반으로 구현된 어플리케이션으로 클라이언트가 웹 브라우저를 이용하여 서비스를 이용할 수 있도록 한다.

2. 회원 관리

(1) 관리자는 회원 목록을 확인하고 회원 정보를 조회, 수정할 수 있어야 한다.

(2) 관리자는 회원의 권한을 조절할 수 있어야 한다.

- 누적 신고 수가 많은 펫시터에게 경고를 줄 수 있어야 한다. 경고가 누적 되면 펫시터 자격을 박탈할 수 있어야 한다.

- 관리자는 회원들의 글쓰기를 차단할 수 있으며, 회원에게 사이트 이용을 제한할 수 있어야 한다.

3. 사이트 기능 관리

(1) 관리자는 사이트 내 모든 게시글을 조회, 삭제할 수 있어야 한다.

(2) 관리자는 유저가 확인할 체크리스트의 기본 값을 설정할 수 있어야 한다.

(3) 관리자는 펫시터 지원자에 대한 지식시험 문제를 관리할 수 있어야 한다.

- 관리자는 유저들이 이용하는 Q&A 게시판에서 질문은 문제로, 답변은 답으로 가공하여 지식시험의 문제로 활용할 수 있어야 한다.

4. 고객과의 상호작용

(1) 관리자는 고객의 문의사항에 응답할 수 있어야 한다.

- 관리자는 유저들의 문의사항 목록을 모아서 볼 수 있어야 한다.

- 관리자는 유저들의 이메일을 참고하여 문의사항에 대한 답변을 보낼 수 있어야 한다.

● 사용자측 요구 분석

1. 기능적인 접근성이 좋아야 한다.

- 사용자는 웹 브라우저를 이용하여 쉽게 사이트에 접근할 수 있어야 한다.
- 사용자는 검색창을 통하여 원하는 펫시터를 빠르게 찾을 수 있어야 한다.
- 사용자는 펫시터의 목록을 거리 순, 이용 순, 평점 순, 조회 순, 즐겨찾기 순으로 정렬할 수 있어야 한다.
- 사용자는 펫시터의 목록을 성별, 펫시팅 동물 종류, 주거환경, 지역, 반려동물 동거 유무를 기준으로 필터링 한 정보를 확인할 수 있어야 한다.

2. 펫시터의 신뢰성이 보장되어야 한다.

- 펫시팅 예약 이후 펫시터의 실제 연락처, 위치정보를 조회할 수 있어야 한다.
- 펫시팅 예약 전 펫시터의 경력 사항, 특이사항 등을 조회할 수 있어야 한다.
- 검사나 시험을 통해 검증되어, 신뢰성 있는 펫시터를 조회할 수 있어야 한다.

3. 예약 전, 펫시팅 환경을 조회할 수 있어야 한다.

- 반려동물을 위탁할 환경을 미리 확인할 수 있어야 한다.
- 위탁 환경에는 펫시터의 가족 구성원, 위탁환경, 다양한 서비스 제공 유무, 펫시터의 반려동물 유무 등을 예로 들 수 있다.

4. 간편하게 펫시팅 서비스를 이용할 수 있어야 한다.

- 기존 예약건을 어플리케이션에서 조회할 수 있고 해당 건에 대해 날짜만 지정하여 간편하게 예약할 수 있어야 한다.
- 예약 정보 입력을 안했을 시, 사이트가 설정한 기본 값으로 정보가 채워지는 서비스가 제공되어야 한다.
- 등록된 사용자의 위치를 자동으로 반영하여 주변에 있는 펫시터 정보를 조회할 수 있어야 한다.

5. 유연한 펫시팅 서비스가 제공되어야 한다.

- 사용 합의 하에 위탁을 종료할 수 있어야 한다.
- 위탁 중 종료할 경우 환불규정에 따라 환불 금액이 계산되어 제공되어야 한다.
- 예약시 위탁 시작 시간과 종료 시간을 설정할 수 있어야 한다.

6. 자신의 회원 정보에 대한 관리가 편리해야 한다.

- 언제든지 자신의 반려동물 정보, 자신의 정보를 수정할 수 있어야 한다.
- 언제든지 회원 탈퇴를 할 수 있어야 한다.

7. 위탁 중 투명성이 보장되어야 한다.

- 체크리스트를 통해 유저가 제시한 요청사항들을 펫시터가 잘 수행하는지 확인할 수 있어야 한다.
- 체크리스트에 등록된 사진, 동영상 등을 통해 체크리스트가 잘 이행되었는지 확인할 수 있어야 한다.
- 사이트 내 문의사항을 언제든지 관리자에게 문의 할 수 있어야 한다.

8. 반려동물 관련 정보를 얻을 수 있어야 한다.

- 반려동물 관련된 지식을 다른 유저와 공유할 수 있어야 한다.

● 펫시터측 요구분석

1. 홍보를 통한 이익이 보장되어야 한다.

- 유저에게 효과적으로 정보를 노출하여 하여 이익으로 직결되는 홍보효과 보장받을 수 있도록 해야 한다.
- 타 유저에게 반려동물 관련 정보제공을 통해 자기 홍보 효과를 누릴 수 있어야 한다.
- 신규 펫시터의 경우에도 어플리케이션을 통해 홍보효과를 누릴 수 있어야 한다.

2. 펫시팅 업무에 대한 기준이 있어야 한다.

- 펫시팅에 익숙하지 못한 신규 펫시터의 경우 펫시팅에 대한 가이드라인이 필요하며 이를 어플리케이션에서 제공할 수 있어야 한다.
- 펫시팅에 필요한 준비물이 무엇인지 어플리케이션을 통해 가이드라인을 제공 받을 수 있어야 한다.

3. 위탁자로부터 부당한 대우를 받지 않아야 한다.

- 투명성 보장을 위해 체크리스트 상에 사진이나 영상을 첨부하여 체크리스트를 잘 이행했음을 입증할 수 있어야 한다.
- 블랙 위탁자의 예약을 거절하거나, 블랙 위탁자를 차단하는 기능을 통해 부당한 대우에 대한 대비를 할 수 있어야 한다.

4. 펫시터의 정보가 지나치게 노출되지 않아야 한다.

- 예약 전에는 펫시터의 정보가 제한적으로 노출되어 개인정보를 보호할 수 있어야 한다.

5. 위탁이 비정상적으로 종료된 경우 정산이 쉬워야 한다.

- 위탁 중 사용자의 요구로 인해 철회될 경우 기회비용에 대한 정산을 어플리케이션에서 펫시터가 설정한 값을 반영하여 계산해 주어야 한다.

6. 위탁 중 갑작스러운 사고에 대처할 수 있어야 한다

- 위급상황시 위탁자에게 위급상황임을 인지시켜줄 수 있는 수단을 어플리케이션에서 제공해야 한다.

7. 펫시터는 자신의 펫시터로서의 정보를 관리할 수 있어야 한다.

- 펫시터는 자신의 홍보효과 상승을 위한 사진 업데이트, 경력 사항 입력, 제공 서비스 등을 언제든지 변경할 수 있어야 한다.

기안서

Location-Based WebApplication

과정명	Java를 활용한 웹 응용 SW 개발자 양성 과정	프로젝트 담당교수	김호진
프로젝트 기간	2019년 12월10일 ~1월10일	작성 날짜	2019년 12월 16일

팀 명	Team Patriots		
프로젝트명	Patriots		
조 장	안준현		
조 원	성열원, 이은채, 박종호, 김현우, 김창기		
프로젝트 주제	누구나 주변 이웃들의 반려동물을 위탁 받을 수 있고, 누구나 간편하게 맡길 수 있는 반려동물 위탁 플랫폼 어플리케이션.		
프로젝트 목표	<p>1. 반려동물과 함께 사는 인구가 늘어나는 요즘, 여행을 가거나 집을 오래 비울 때 반려동물을 맡길 곳이 필요하다. 특히 주변에 지인이 없을 경우 곤란한 사람들의 경우 위탁이 쉽지 않다. 이러한 수요를 파악하여 위치 기반 펫시터들의 정보 제공을 통해 누구나 쉽게 근처에 있는 펫시터에게 반려동물을 위탁할 수 있는 플랫폼을 제공하는 어플리케이션을 개발하고자 한다.</p> <p>2. 반려동물 관련 직업들이 늘어나는 만큼, 펫시팅에 대한 관심도도 높아지고 있다. 하지만 펫시팅을 진행하기 위해서는 특정 사이트에서 까다로운 과정을 통과해야하기에 접근성이 높지 못하다. 이를 착안하여 누구나 펫시터를 쉽게 지원할 수 있고, 위탁을 맡길 수 있는 플랫폼을 제공하는 어플리케이션을 개발하고자 한다.</p>		
사용 기술	Apach Tomcat, Oracle, Ajax, javascript, HTML/CSS, XML, Java, My Batis, Bootstrap		
구현 내용	<ul style="list-style-type: none"> • Web 구현 <ol style="list-style-type: none"> 1. 메인페이지 <ul style="list-style-type: none"> - 위치, 위탁날짜, 등의 정보를 통해 펫시터 찾기 기능 제공 2. 마이페이지 <ul style="list-style-type: none"> - 회원 정보 수정, 회원탈퇴, 펫시터 지원, 위탁 및 예약 정보를 열람하는 기능을 제공 3. 펫시터 페이지 <ul style="list-style-type: none"> - 펫시팅 정보 수정, 업적 관리, 예약 및 위탁 관리의 기능을 제공 4. 관리자 페이지 <ul style="list-style-type: none"> - 회원 정보 관리, 시험 문제 관리, 문의사항 열람의 기능을 제공 5. 펫시터 찾기 페이지 <ul style="list-style-type: none"> - 특정 필터를 통해 펫시터 목록을 출력해주는 기능을 제공 6. 공지사항 <ul style="list-style-type: none"> - 어플리케이션의 공지사항을 열람할 수 있는 기능을 제공 7. QnA 페이지 <ul style="list-style-type: none"> - 사용자간 질문과 답변을 통해 반려동물 관련 지식을 얻을 수 있는 기능을 제공 		
프로젝트 일정	1주차	모델 사이트 분석	
	2~3주차	화면설계, 기능 설계, DB설계	
	3주차	화면 디자인, 개발	
	3~4주차	개발, 개별 업무 통합	
업무 분담	안준현	기획, 업무 분담, DB 모델링 및 구축 MVC 패턴 구현	
	성열원	DB 모델링 및 관리, MVC 패턴 구현	
	이은채	뷰페이지 작성, MVC 패턴 구현	
	박종호	MVC 패턴 구현, DB 모델링 및 관리	
	김현우	뷰페이지 작성, MVC 패턴 구현	
	김창기	뷰페이지 작성, MVC 패턴 구현	

배경지식

Location-Based WebApplication

JAVA 란?

자바(Java) 는 썬 마이크로시스템즈의 제임스 고슬링(James Gosling)과 다른 연구원들이 개발한 객체 지향적 프로그래밍 언어이다.

JAVA 의 핵심 목표

- 객체 지향 방법론을 사용해야 한다.
- 같은 프로그램(바이트코드)이 여러 운영 체제(마이크로프로세서)에서 실행될 수 있어야 한다.
- 컴퓨터 네트워크 접근 기능이 기본으로 탑재되어 있어야 한다.
- 원격 코드를 안전하게 실행할 수 있어야 한다.
- 다른 객체 지향 언어들의 좋은 부분만 가지고 와서 사용하기 편해야 한다.

JAVA 의 간단 용어정리

Class 클래스

클래스란? 객체를 정의해 놓은 것

클래스의 용도 : 객체를 생성하기 위해 사용 됩니다.

클래스는 객체를 생성하는데 사용 될뿐 그 자체는 아닙니다.

Object 객체

객체란? 실제로 존재하는 것. 사물 또는 개념, 사전적의미 : 물건, 물체, 목적(물)

포괄적으로, 모든 사물이나 개념 을 말합니다. ex) 모니터,키보드,바나나,자동차 등을 모두 객체 라고 부릅니다.

객체의 용도 : 객체의 속성과 기능에 따라 다릅니다.

Instance 인스턴스

인스턴스란? 의미상으로는 객체랑 비슷함, 객체(Object)는 인스턴스(Instance) 를 포함

인스턴스화란? 클래스로부터 객체를 만드는과정을 클래스의 인스턴화 라고 합니다. // 인스턴스 생성법 : Car a = new Car();

클래스로부터 만들어진 객체를 그 클래스의 인스턴스(객체) 라고 합니다.

Oracle SQL developer

- Database 개발을 위한 그래픽 인터페이스로써 SQL과 PL/SQL 구문 실행, SQL 스크립트 실행, PL/SQL 구문 편집 및 디버깅 기능을 갖추고 있는 Tool입니다.

SQL

SQL은 관계형 데이터베이스 관리 시스템의 데이터를 관리하기 위해 설계된 특수 목적의 프로그래밍 언어이다. 관계형 데이터베이스 관리 시스템에서 자료의 검색과 관리, 데이터베이스 스키마 생성과 수정, 데이터베이스 객체 접근 조정 관리를 위해 고안되었다.

Spring MVC

1. 웹 개발을 위한 MVC 패턴이 적용된 Spring Framework.

Spring 에서는 DispatcherServlet 클래스가 컨트롤러 역할을 수행하게 된다.

2. Spring MVC 의 주요 구성요소

- DispatcherServlet
 - : 클라이언트의 요청을 전달받는다.
 - Controller(사용자 정의 컨트롤러)에게 클라이언트의 요청을 전달하고,
 - Controller 가 리턴한 결과값을 View 에 전달하여
 - 적합한 응답을 생성하도록 처리한다.
- HandlerMapping
 - : 클라이언트의 요청 URL을 어떤 Controller 가 처리할지 결정한다
- Controller
 - : 사용자 정의 Controller. 액션 처리. 우리가 작성
- ModelAndView
 - : View 객체 정보와 결과값이 들어있는 객체
- ViewResolver
 - : 요청에 대한 처리 결과를 보여줄 View 를 결정한다
- View
 - : 요청에 대한 처리 결과를 보여줄 응답(결과 출력)을 생성한다. 우리가 작성.
- web.xml
 - : DispatcherServlet 서블릿 객체 등록. 우리가 작성
- dispatcher-servlet.xml
 - : Controller 객체 등록. 우리가 작성

3. Spring MVC 실행 흐름

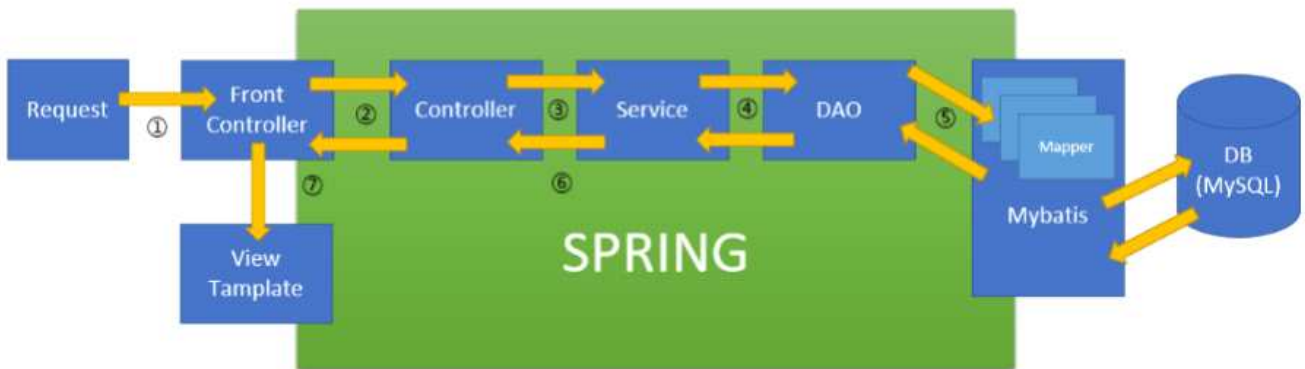
- 사용자 요청 → DispatcherServlet
- DispatcherServlet → HandlerMapping 에 주소 분석 요청.
- DispatcherServlet → 특정 Controller 객체 호출 및 결과 수신.
 - 결과는 ModelAndView 객체.
 - ※ ModelAndView 객체
 - : View 객체 정보와 결과값이 들어있는 객체.
- DispatcherServlet → ViewResolver 에 View 객체 분석 요청.
- DispatcherServlet → 특정 View 객체에 Model 객체를 넘겨주고 결과 출력.

4. Spring MVC 에서 우리가 해야 할 일

- Controller
 - 액션 처리를 담당하는 Controller 클래스 작성.
 - Annotation 을 사용하거나 Controller 인터페이스를 구현하여 작성.
- View
 - 결과 출력용 JSP 페이지 작성.
- web.xml
 - 서블릿 주소와 DispatcherServlet 클래스 등록.
- dispatcher-servlet.xml
 - 우리가 작성한 Controller 객체 등록.
 - URL 매핑 주소 등록.

Spring MVC 구조

Spring MVC의 요청에 대한 흐름



1. 사용자의 요청을 Front Controller에 전달
2. 전달된 요청은 적절한 Controller를 찾아 호출
3. Controller는 해당 Service 객체를 찾아 호출
4. Service객체는 DAO객체를 이용하여 원하는 Data를 요청
5. DAO객체는 MyBatis를 이용하는 Mapper를 통해 작업을 수행(CRUD)
6. Service를 통해 처리한 데이터를 Controller에게 전달
7. Controller는 다시 스프링 MVC쪽으로 데이터를 전달

SpringMVC가 처리하는 작업	개발자가 직접 해야하는 작업
URI를 분석해서 적절한 Controller를 찾는 작업	특정URI에 동작하는 Controller를 설계하는 작업
Controller에 필요한 메서드를 호출하는 작업	Service / DAO 객체 생성
Controller의 결과 데이터를 View로 전달하는 작업	Controller 내에 원하는 결과를 메서드로 설계
적절한 View를 찾는 작업	View에서 전달받은 데이터의 출력

Spring Annotation

Java에서 어노테이션(Annotation)이란?

자바 개발을 하다 보면 클래스 또는 메서드 또는 변수 앞에 @Override 같은 @ 표시는 어노테이션이라고 부른다. 이 어노테이션은 JEE5(Java Platform, Enterprise Edition 5)부터 새롭게 추가된 요소입니다.

이 어노테이션으로 인해 데이터의 유효성 검사 등을 쉽게 알 수 있고, 이와 관련한 코드가 깔끔해지게 된다. 일단 어노테이션의 용도는 다양한 목적이 있지만 메타 데이터의 비중이 가장 크다 할 수 있습니다.

Annotation 종류

애너테이션	기능
@Controller	Controller 객체임을 명시
@Repository	DAO 객체임을 명시
@Service	Service 객체임을 명시
@RequestMapping	특정 URI에 매칭되는 클래스나, 메서드임을 명시
@RequestParam	요청에서 특정한 파라미터 값을 찾아낼 때 사용
@RequestHeader	요청에서 특정 HTTP 헤더 정보를 추출할 때 사용
@PathVariable	현재의 URI에서 원하는 정보를 추출할 때 사용
@CookieValue	현재 사용자의 쿠키가 존재하는 경우 쿠키의 이름을 이용해서 쿠키 값을 추출
@ModelAttribute	자동으로 해당 객체를 뷰까지 전달하도록 만드는 애너테이션
@SessionAttribute	세션상에서 모델의 정보를 유지하고 싶은 경우 사용
@InitBinder	파라미터를 수집해서 객체로 만들 경우
@ResponseBody	리턴타입이 HTTP의 응답메시지로 전송
@RequestBody	요청 문자열이 그대로 파라미터로 전달

Mybatis란?

마이바티스란?

- mybatis 는 persistence(퍼시스턴스: 영속적인: 소멸되지 않고 오래도록 지속되는) 계층의 프레임워크로 다른 객체 관계 매핑 프레임워크들과는 달리 정형화된 테이블 구조에 의존하지 않고 SQL 구문의 실행 결과에 객체를 매핑하는 방식으로 테이블 구조가 복잡해지고 정규화 되어 있지 않더라도 테이블을 변경할 필요 없이 기존 개발 방법에서 사용한 SQL 을 그대로 사용할 수가 있으며, SQL 에 적용한 각각의 튜닝 항목들도 모두 그대로 유지된다.

마이바티스의 구성

- mybatis 프로젝트는 mybatis SQL Map 과 mybatis DAO 컴포넌트로 구성되어 있다. 이 중, mybatis SQL Map 은 SQL 문 관리에 초점을 맞추고 있다. mybatis DAO 컴포넌트는 hibernate 와 mybatis SQL Map 을 함께 사용할 경우 트랜잭션 통합과 같은 저장 매커니즘에 초점을 맞추고 있다.

Mybatis의 특징

- 프레임워크(환경, 틀, 약속)
 - mybatis 는 JDBC 프레임워크다.
 - 개발자는 SQL 을 작성하기로 약속하고, 마이바티스는 JDBC 를 사용해서 실행하기로 약속한다.
- 간단함
 - 간단한 퍼시스턴스 프레임워크
 - JDBC 를 사용할 try / catch / finally 구문을 사용할 필요가 없다.
- 생산성
 - JDBC 와 비교했을 때 62% 정도 줄어드는 코드와 간단한 설정 (근거 : 이탈리아 자바 사용자 그룹의 연구)
- 성능
 - 구조적 강점(데이터 접근 속도를 높여주는 join 매핑)
 - 여러가지 방식의 데이터 가져오기 전략(가져오기 미루기, SQL 줄이기 기법 등)
- 관심사의 분리
 - 설계를 향상(차후 유지보수성을 위해)
 - 리소스를 관리하여 계층화를 지원, (Connection, PreparedStatement, ResultSet)
- SQL 문이 애플리케이션 소스 코드로부터 완전히 분리
- 작업의 분배
 - 팀을 세분화하여 업무 구분을 명확히 하는 것에 도움을 준다.
- 이식성
 - 어떤 프로그래밍 언어로도 구현이 가능하다. (JAVA, C#, RUBY 등)
- 오픈소스
 - 무료 + 커뮤니티
- SQL Mapper
 - ORM 은 데이터베이스 객체를 자바 객체로 매핑함으로써 객체 간의 관계를 바탕으로 SQL 을 자동으로 생성해 주지만, 마이바티스는 SQL 을 작성 및 명시해 주어야 한다.
- 스프링과 같은 외부 트랜잭션 관리자를 사용할 수도 있다.
- 스프링 연동 모듈을 제공하기 때문에 스프링과 연동할 수도 있다.

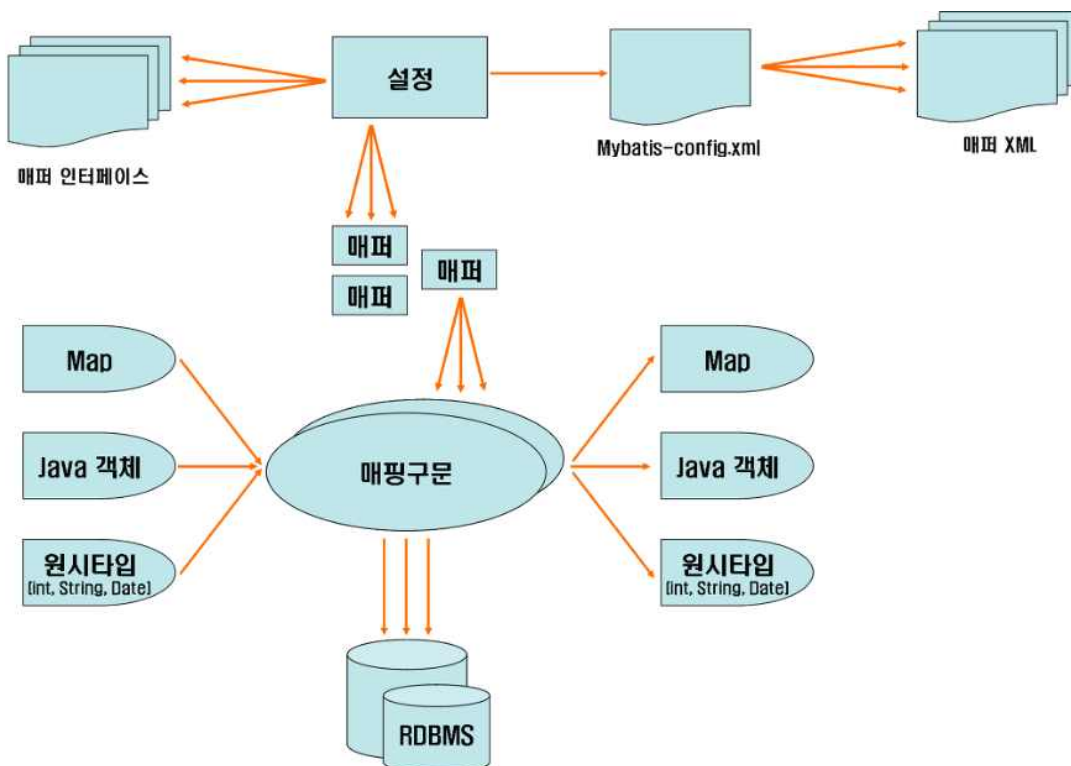
Mybatis의 작동 원리

- 1) 파라미터(자바 빈즈, Map 또는 원시 래퍼)로써 객체를 제공한다.
파라미터 객체는 update 문 내부에 입력 값을 세팅하기 위해 사용되거나 쿼리문의 where 절을 세팅하기 위해 사용된다.
- 2) 매핑된 statement 를 실행한다.
Data Mapper 프레임워크는 PreparedStatement 인스턴스를 생성하고 제공된 파라미터 객체를 사용해서 파라미터를 세팅한다.
그리고 statement 를 실행하고 ResultSet 으로부터 결과 객체를 생성한다.
- 3) update 의 경우에 영향을 미친 rows 의 숫자를 반환한다.
조회문일 경우 한 개(single)의 객체 또는 컬렉션 객체를 반환한다.
파라미터처럼 결과 객체는 자바 빈즈, Map, 원시 래퍼, XML 이 될 수 있다.

Mybatis의 작동 흐름

- 데이터 처리를 위해 DAO 는 mybatis 에 제공하는 객체의 메소드를 호출한다.
- mybatis 는 SQL 문이 저장된 매퍼 파일에서 데이터 처리에 필요한 SQL 문을 찾는다.
- mybatis 는 매퍼 파일에서 찾은 SQL을 서버로 보내고자 JDBC 드라이버를 사용한다.
- JDBC 드라이버는 SQL 문을 데이터베이스 서버로 전송한다.
- mybatis 는 select 문의 실행 결과를 값 객체에 담아서 반환한다.
(insert, update, delete 문인 경우 입력, 변경, 삭제된 레코드의 갯수를 반환한다.)

Mybatis의 구조



- mybatis-config 는 mybatis의 메인 환경설정 파일이다. 어떤 DBMS와 커넥션 맺을지, 어떤 Mapper 파일들이 있는지 등 알 수 있다.
- Mybatis는 Mapper 파일에 있는 각 SQL 명령어들을 Map 에 담아서 저장하고 관리한다.

Mybatis의 관련 사이트

- ibatis 이전 홈페이지 <http://ibatis.apache.org/>
- mybatis 새 홈페이지 <http://blog.mybatis.org/>
- 다운로드
 - mybatis <http://blog.mybatis.org/p/products.html>
 - spring 연동 <http://github.com/mybatis/spring/release/>
- 도움말 <http://mybatis.github.io/mybatis-3/ko/getting-started.html>
- API <http://ibatis.apache.org/docs/java/dev/>

Mybatis의 Mapper XML 파일

mybatis 의 가장 큰 장점은 매핑된 구문이며, SQL Map XML 파일은 상대적으로 간단하다.
그리고 동일한 기능의 JDBC 코드와 비교하면 다량의 코드가 감소하기도 한다.
즉, mybatis 는 SQL 을 작성하고 관리하는데 집중하도록 만들어졌다.

※ SQL Map XML 파일은 첫 번째(first class) 요소만을 가진다.

- cache
 - : 해당 명명공간을 위한 캐시 설정
- cache-ref
 - : 다른 명명공간의 캐시 설정에 대한 참조
- resultMap
 - : 데이터베이스 결과데이터를 객체에 로드하는 방법을 정의하는 요소
- parameterMap
 - : 권장하지 않음,
예전에는 파라미터를 매핑하기 위해 사용하였으나 현재는 사용하지 않음.
- sql
 - : 다른 구문에서 재사용하기 위한 SQL 조각
- insert
 - : 매핑된 insert 구문
- update
 - : 매핑된 update 구문
- delete
 - : 매핑된 delete 구문
- select
 - : 매핑된 select 구문

JavaScript 개요

컴파일 과정이 필요 없거나 처리 과정이 적은 프로그램에 유리한 스크립트 언어로
다른 언어에 비해 개발 시간이 짧고, 소스 코드의 수정 및 유지보수가 간편하다.
HTML 코딩에서 자바 스크립트 소스의 명시만으로 웹 브라우저가 자동 실행된다.

자바스크립트는 웹 페이지에서 가장 많이 사용되는 스크립팅(Scripting) 언어로
스크립팅(Scripting) 언어는 크게 나누면
자바스크립트나 DHTML, Visual Basic 등과 같은 브라우저 스크립팅(Browser Scripting)
JSP, ASP, PHP 와 같은 서버 스크립팅(Server Scripting) 언어로 구분할 수 있다.

특히 자바스크립트는 서버 스크립팅(Server Scripting) 언어와 연동되어
입력란에 입력한 값이 제대로 된 값인지의 여부 등을 체크하는
상호작용(Interactive)적 요소와
웹 페이지에 동적(Dynamic)인 효과 등을 적용하는 부분에 사용되면
그 밖에도 많은 기능을 사용자가 추가하여 사용하는 매우 중요한 요소라 할 수 있다.

※ Javascript 는 HTML 문서에
상호작용(Interactive)하는 요소와 동적인(Dynamic)인 요소를
추가하기 위해 사용하는 언어이다.

JQuery 개요

○ 개요

- jQuery 는 빠르고 간결한 Javascript Library
- HTML Document traversing, 이벤트 처리, 애니메이션, Ajax 를 단순화하여 빠른 웹 개발을 가능하게 한다.
- CSS 규격 지원 : CSS 1 ~ 3 와 기본 XPath 지원.
- 다양한 브라우저 지원.
- 경량 파일(약 90kb 수준의 파일 크기)

JQuery 의 사용방법

1. <http://jquery.com> 로 부터 파일을 다운로드해서 사용하는 방법

※ 다운로드 받을 수 있는 종류

- 『production』 버전
: 소스를 압축해서(불필요한 공백 및 개행 제거)
웹 서버에서 빠르게 실행할 수 있도록 경량화/최소화 한 버전.
- 『development』 버전
: 테스트나 디버깅 또는 코드에 대한 분석을 위해
압축을 하지 않고 가독성을 높여놓은 버전
(공백 및 개행 처리가 되어 있음)

2. CDN(Content Delivery Network)을 통해 사용하는 방법

※ CDN : 웹 페이지를 실행할 때 마다
공개 서버에서 네트워크를 통해
jQuery 를 다운로드 받을 수 있게 해주는 개념

별도의 파일을 다운로드 받을 필요 없이

『<script type = "text/javascript" src = "http://code.jquery.com/jquery-1.12.4.min.js"></script>』
와 같은 구문을 통해 포함시킨다

JQuery 대표적인 선택자

- 타입 선택자(요소 선택자 : 엘리먼트 선택자)
: 페이지 상의 모든 <p> 요소를 선택하고자 할 경우 다음과 같이 기술하면 된다.
『\$("p")』
- 클래스 선택자
: 『<p class = "test">』라는 클래스가 test 인 요소를 선택하고자 할 경우
다음과 같이 기술하면 된다.
『\$(".test")』
- 아이디 선택자
: 『<p id = "exam">』 라는 아이디가 exam 인 요소를 선택하고자 할 경우
다음과 같이 기술하면 된다.
『\$("#exam")』

**DB Modeling
&
테이블 명세서**

Location-Based WebApplication

(1) 회원 관련 테이블

테이블 명		SIGNUP		테이블 정보	회원 등록		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	USER_NUM	유저번호	N/A	NUMBER	Not Null		PK
2	PWD	패스워드	N/A	VARCHAR2(40)	Not Null		
3	USERID	ID	N/A	VARCHAR2(40)	Not Null		
4	USERNAME	이름	N/A	VARCHAR2(40)	Not Null		
5	SSN	주민번호	N/A	VARCHAR2(40)	Not Null		
6	ZIPCODE	우편번호	N/A	VARCHAR2(10)	Not Null		
7	ADDRESS	주소	N/A	VARCHAR2(100)	Not Null		
8	ADDRDETAILS	상세주소	N/A	VARCHAR2(100)			
9	ADDRLAT	주소 위도	N/A	NUMBER	Not Null		
10	ADDRLONG	주소 경도	N/A	NUMBER	Not Null		
11	TEL	전화번호	N/A	VARCHAR2(40)	Not Null		
12	EMAIL	E-Mail	N/A	VARCHAR2(100)	Not Null		
13	NICKNAME	닉네임	N/A	VARCHAR2(20)	Not Null		
14	SU_DATE	가입일자	N/A	DATE	Not Null	SYSDATE	

테이블 명		FAVORITES		테이블 정보	찜		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	FAV_NUM	찜 번호	N/A	NUMBER	Not Null		PK
2	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK
3	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		FK
4	ISUDATE	찜일자	N/A	DATE	Not Null		

테이블 명		WITHDRAWAL		테이블 정보	회원 탈퇴		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	WD_NUM	탈퇴 번호	N/A	NUMBER	Not Null		PK
2	ISUDATE	탈퇴 일자	N/A	DATE	Not Null	SYSDATE	
3	WD_R_NUM	탈퇴 사유 번호	N/A	NUMBER	Not Null		FK
4	DETAILS	탈퇴 사유 자세히	N/A	VARCHAR2(1000)			
5	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK

테이블 명		WD_REASON		테이블 정보	탈퇴 사유 작성		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	WD_R_NUM	탈퇴 사유 번호	N/A	NUMBER	Not Null		PK
2	REASON	탈퇴 사유	N/A	VARCHAR2(200)	Not Null		

→ 다음 페이지에 계속

(1) 회원 관련 테이블

테이블 명		MESSAGE		테이블 정보	쪽지 발신		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	MSG_NUM	발신 번호	N/A	NUMBER	Not Null		PK
2	MSG_SND_USER	발신자	N/A	NUMBER	Not Null		FK
3	MSG_RCV_USER	수신자	N/A	NUMBER	Not Null		FK
4	MAINTTEXT	내용	N/A	VARCHAR2(1000)	Not Null		
5	ISUDATE	발신일	N/A	DATE	Not Null		
6	SND_DELDATE	발신함 삭제일	N/A	DATE			
7	RCV_DELDATE	수신함 삭제일	N/A	DATE			

(2) 관리자 관련 테이블

테이블 명		REGISTER_ADMIN		테이블 정보	관리자		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	ADMIN_NUM	관리자번호	N/A	NUMBER	Not Null		PK
2	ID	ID	N/A	VARCHAR2(16)	Not Null		
3	PW	비밀번호	N/A	VARCHAR2(16)	Not Null		

(3) 펫시터 관련 테이블

테이블 명		PST_SHIFT		테이블 정보	펫시터 전환		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		PFK
2	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK

테이블 명		B_TYPE		테이블 정보	건물 유형		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	B_TYPE_NUM	건물 유형 번호	N/A	NUMBER	Not Null		PK
2	B_TYPE	건물 유형	N/A	VARCHAR2(20)	Not Null		UK

테이블 명		MEDICATION		테이블 정보	투약 가능 여부		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	MEDICATION	번호	N/A	NUMBER			PK
2	OK	투약 가능 여부	N/A	VARCHAR2(10)	Not Null		CK

테이블 명		PICK_UP		테이블 정보	픽업 서비스 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	PICK_UP	번호	N/A	NUMBER			PK
2	OK	투약 가능 여부	N/A	VARCHAR2(10)	Not Null		CK

→ 다음 페이지에 계속

(3) 펫시터 관련 테이블

테이블 명		GARDEN		테이블 정보	마당 서비스 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	GARDEN	번호	N/A	NUMBER			PK
2	OK	마당 서비스 여부	N/A	VARCHAR2(10)	Not Null		CK

테이블 명		FAMILY		테이블 정보	가족 동거 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	FAMILY	번호	N/A	NUMBER			PK
2	OK	가족 동거 여부	N/A	VARCHAR2(10)	Not Null		CK

테이블 명		KIDS		테이블 정보	픽업 서비스 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	KIDS	번호	N/A	NUMBER			PK
2	OK	투약 가능 여부	N/A	VARCHAR2(10)	Not Null		CK

테이블 명		OTHER_ANIMAL		테이블 정보	다른 동물 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	OTHER_ANIMAL	번호	N/A	NUMBER			PK
2	OK	다른 동물 유무	N/A	VARCHAR2(10)	Not Null		CK

테이블 명		SNACK		테이블 정보	간식 제공 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	SNACK	번호	N/A	NUMBER			PK
2	OK	간식 제공 여부	N/A	VARCHAR2(10)	Not Null		CK

테이블 명		WALK		테이블 정보	산책 서비스 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	WALK	번호	N/A	NUMBER			PK
2	OK	산책 서비스 여부	N/A	VARCHAR2(10)	Not Null		CK

테이블 명		BEAUTY		테이블 정보	미용 서비스 유무		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	BEAUTY	번호	N/A	NUMBER			PK
2	OK	미용 서비스 여부	N/A	VARCHAR2(10)	Not Null		CK

→ 다음 페이지에 계속

(3) 펫시터 관련 테이블

테이블 명		PST_PROFILE		테이블 정보	프로필 등록		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	P_S_P_NUM	프로필 등록 번호	N/A	NUMBER	Not Null		PK
2	CHECKIN_START	표준 체크인시작시간	N/A	VARCHAR2(10)	Not Null		
3	CHECKIN_END	표준 체크인끝시간	N/A	VARCHAR2(10)	Not Null		
4	CHECKOUT_START	표준 체크아웃시작시간	N/A	VARCHAR2(10)	Not Null		
5	CHECKOUT_END	표준 체크아웃끝시간	N/A	VARCHAR2(10)	Not Null		
6	PRICE	주간 가격 금액	N/A	NUMBER	Not Null		
7	RECORDS	이력사항	N/A	VARCHAR2(2000)	Not Null		
8	INTRODUCE	소개	N/A	VARCHAR2(1000)	Not Null		
9	BEAUTY	미용 서비스 유무	N/A	NUMBER	Not Null		FK
10	WALK	산책 서비스 유무	N/A	NUMBER	Not Null		FK
11	SNACK	간식 제공 여부	N/A	NUMBER	Not Null		FK
12	OTHER_ANIMAL	다른 동물 유무	N/A	NUMBER	Not Null		FK
13	FAMILY	가족 동거 유무	N/A	NUMBER	Not Null		FK
14	KIDS	14세 미만 아동 유무	N/A	NUMBER	Not Null		FK
15	GARDEN	마당 유무	N/A	NUMBER	Not Null		FK
16	PICK_UP	픽업 서비스 유무	N/A	NUMBER	Not Null		FK
17	MEDICATION	투약 가능 여부	N/A	NUMBER	Not Null		FK
18	MAX_SITABLE	최대 위탁 가능 개체	N/A	NUMBER	Not Null		FK
19	B_TYPE_NUM	건물 유형 번호	N/A	NUMBER	Not Null		FK
20	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		FK

(4) 업적 관련 테이블

테이블 명		QUEST_CPL		테이블 정보	업적 달성		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	QUEST_NUM	업적 번호	N/A	NUMBER	Not Null		PFK
2	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		PFK
3	ISUDATE	달성일자	N/A	DATE	Not Null		
4	USE	사용여부	N/A	NUMBER	Not Null		

테이블 명		QUEST		테이블 정보	업적		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	QUEST_NUM	업적 번호	N/A	NUMBER	Not Null		PK
2	LINK	아이템 링크	N/A	VARCHAR2(200)	Not Null		
3	Q_NAME	업적명	N/A	VARCHAR2(50)	Not Null		
4	I_NAME	아이템명	N/A	VARCHAR2(50)	Not Null		
5	CONDITION	달성조건	N/A	VARCHAR2(50)	Not Null		

→ 다음 페이지에 계속

(5) 체크리스트 관련 테이블

테이블 명		CCL_REQUEST		테이블 정보	요청 체크리스트 항목 작성		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	C_REQ_NUM	요청 중 체크리스트 항목 번호	N/A	NUMBER	Not Null		PK
2	STTDATE	시작시간	N/A	DATE	Not Null		
3	ENDDATE	종료시간	N/A	DATE	Not Null		
4	MAINTTEXT	내용	N/A	VARCHAR2(200)	Not Null		
5	R_U_NUM	예약 정보 작성(전) 번호	N/A	NUMBER	Not Null		FK
6	R_A_NUM	예약 정보 작성(후) 번호	N/A	NUMBER	Not Null		FK
7	A_T_NUM	승인 여부 번호	N/A	NUMBER	Not Null		FK

테이블 명		CCL_APPROVE		테이블 정보	체크리스트 승인		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	C_A_NUM	체크리스트 승인 번호	N/A	NUMBER	Not Null		PK
2	C_REQ_NUM	요청 중 체크리스트 항목 번호	N/A	NUMBER	Not Null		FK
3	ISUDATE	승인 시간	N/A	DATE	Not Null		
4	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		FK

테이블 명		CCL_CONFIRM		테이블 정보	확정 체크리스트 항목 작성		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	C_C_NUM	확정 체크리스트 항목 번호	N/A	NUMBER	Not Null		PK
2	EXCUTE	실행여부	N/A	CHAR(1)	Not Null		
3	TIME	실행시각	N/A	DATE	Not Null		
4	VIDLINK	동영상 주소	N/A	VARCHAR2(200)	Not Null		
5	IMGLINK	사진 주소	N/A	VARCHAR2(200)	Not Null		
6	S_A_NUM	위탁 시작 번호	N/A	NUMBER	Not Null		FK
7	C_A_NUM	체크리스트 승인 번호	N/A	NUMBER	Not Null		FK

테이블 명		CCL_FINAL		테이블 정보	최종 체크리스트 항목 작성		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	C_F_NUM	최종 체크리스트 항목 번호	N/A	NUMBER	Not Null		PK
2	C_C_NUM	확정 체크리스트 항목 번호	N/A	NUMBER	Not Null		FK
3	S_C_NUM	위탁완료번호	N/A	NUMBER	Not Null		FK

→ 다음 페이지에 계속

(5) 체크리스트 관련 테이블

테이블 명		APPROVE_TYPE		테이블 정보	승인 여부		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	A_T_NUM	승인 여부 번호	N/A	NUMBER	Not Null		PK
2	TYPE	상태값	N/A	VARCHAR2(15)	Not Null		

테이블 명		CCL_D		테이블 정보	체크리스트 기본값		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	C_D_NUM	체크리스트 기본값 번호	N/A	NUMBER	Not Null		PK
2	ADMIN_NUM	관리자번호	N/A	NUMBER	Not Null		FK
3	P_T_NUM	타입 번호	N/A	NUMBER	Not Null		FK

테이블 명		CCL_D_R		테이블 정보	체크리스트 기본값 항목 등록		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	C_D_R_NUM	기본값 항목 등록 번호	N/A	NUMBER	Not Null		PK
2	C_D_NUM	체크리스트 기본값 번호	N/A	NUMBER	Not Null		FK
3	MAINTXT	내용	N/A	VARCHAR2(200)	Not Null		

테이블 명		REG_MY_CCL		테이블 정보	사용자 저장 체크리스트		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	R_M_C_NUM	사용자 저장 체크리스트	N/A	NUMBER	Not Null		PK
2	P_R_NUM	반려동물 등록 번호	N/A	NUMBER	Not Null		FK
3	STTIME	시작시간	N/A	VARCHAR2(15)	Not Null		
4	ENDTIME	끝시간	N/A	VARCHAR2(15)	Not Null		CK
5	MAINTXT	내용	N/A	VARCHAR2(200)	Not Null		

(6) 공지사항 관련 테이블

테이블 명		NOTICE		테이블 정보	공지사항		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	NTC_NUM	공지글번호	N/A	NUMBER	Not Null		PK
2	ADMIN_NUM	작성자번호	N/A	NUMBER	Not Null		FK
3	SUBJECT	공지글제목	N/A	VARCHAR2(100)	Not Null		
4	VIEWS	조회수	N/A	NUMBER	Not Null		
5	ISUDATE	작성일	N/A	DATE	Not Null		
6	MAINTXT	공지글내용	N/A	VARCHAR2(1000)	Not Null		
7	DELDATE	공지 삭제 일자	N/A	DATE			

→ 다음 페이지에 계속

(7) Q/A 관련 테이블

테이블 명		QUESTION		테이블 정보	질문		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	Q_NUM	질문번호	N/A	NUMBER	Not Null		PK
2	MAINTEXT	질문내용	N/A	VARCHAR2(3000)	Not Null		
3	ISUDATE	질문시간	N/A	DATE	Not Null		
4	USER_NUM	유저번호	N/A	NUMBER	Not Null		
5	VIEWS	조회 수	N/A	NUMBER	Not Null		
6	DEL_DATE	질문 삭제 일자	N/A	DATE			

테이블 명		Q_COMMENT		테이블 정보	댓글 및 대댓글		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	Q_C_NUM	질문 댓글 번호	N/A	NUMBER	Not Null		PK
2	Q_NUM	질문번호	N/A	NUMBER	Not Null		FK
3	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK
4	MAINTEXT	내용	N/A	VARCHAR2(600)	Not Null		
5	Q_C_PARENT_NUM	부모 댓글 번호	N/A	NUMBER			FK
6	ISUDATE	댓글 작성일	N/A	DATE	Not Null		
7	DELDATE	질문 댓글 삭제 일자	N/A	DATE			

테이블 명		ANSWER		테이블 정보	답변		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	A_NUM	답변 번호	N/A	NUMBER	Not Null		PK
2	ISUDATE	답변시간	N/A	DATE	Not Null		
3	MAINTEXT	답변내용	N/A	VARCHAR2(3000)	Not Null		
4	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK
5	Q_NUM	질문번호	N/A	NUMBER	Not Null		FK
6	DELDATE	답변 삭제 일자	N/A	DATE			

(8) 신고 관련 테이블

테이블 명		REPORT		테이블 정보	신고		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	RPT_NUM	신고 번호	N/A	NUMBER	Not Null		PK
2	REPORTER	신고인	N/A	NUMBER	Not Null		FK
3	REPORTED	피신고인	N/A	NUMBER	Not Null		FK
4	ISUDATE	신고일자	N/A	DATE	Not Null		
5	DETAILS	신고내용	N/A	VARCHAR2(1000)	Not Null		
6	PRT_TYPE_NUM	신고 구분 번호	N/A	NUMBER	Not Null		FK

→ 다음 페이지에 계속

(8) 신고 관련 테이블

테이블 명		RPT_TYPE		테이블 정보	신고 유형 구분		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	PRT_TYPE_NUM	신고 구분 번호	N/A	NUMBER	Not Null		PK
2	TYPE	신고 내용	N/A	VARCHAR2(100)	Not Null		

테이블 명		SANCTIONS		테이블 정보	신고 처리		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	SCT_NUM	신고 처리 번호	N/A	NUMBER	Not Null		PK
2	ADMIN_NUM	관리자번호	N/A	NUMBER	Not Null		FK
3	RPT_NUM	신고 번호	N/A	NUMBER	Not Null		FK
4	SCT_TYPE_NUM	신고 처리 종류 번호	N/A	NUMBER	Not Null		FK
5	SCTDAY	일수	N/A	NUMBER			
6	ISUDATE	신고처리 일자	N/A	DATE	Not Null		

테이블 명		SCT_TYPE		테이블 정보	신고 처리 종류		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	SCT_TYPE_NUM	신고 처리 종류 번호	N/A	NUMBER	Not Null		PK
2	TYPE	신고 처리 종류	N/A	VARCHAR2(20)	Not Null		

(9) 문의 관련 테이블

테이블 명		MTOM_Q		테이블 정보	1:1문의		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	M_Q_NUM	문의번호	N/A	NUMBER	Not Null		PK
2	VIEWS	조회 수	N/A	NUMBER	Not Null		
3	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK
4	ISUDATE	문의날짜	N/A	DATE	Not Null		
5	MAINTTEXT	문의내용	N/A	VARCHAR2(1000)	Not Null		
6	M_Q_T_NUM	문의분류번호	N/A	NUMBER	Not Null		FK
7	DELDATE	문의 삭제 일자	N/A	DATE			

테이블 명		MTOM_Q_TYPE		테이블 정보	1:1문의 분류		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	M_Q_T_NUM	문의분류번호	N/A	NUMBER	Not Null		PK
2	TYPE	문의분류	N/A	VARCHAR2(20)	Not Null		

(9) 문의 관련 테이블

테이블 명		MTOM_A		테이블 정보	1:1문의 답글		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	M_A_NUM	답글번호	N/A	NUMBER	Not Null		PK
2	ADMIN_NUM	관리자번호	N/A	NUMBER	Not Null		FK
3	MAINTTEXT	답글내용	N/A	VARCHAR2(1000)	Not Null		
4	M_Q_NUM	문의번호	N/A	NUMBER	Not Null		FK
5	DELDATE	답글 삭제 일자	N/A	DATE			
6	ISUDATE	답변일자	N/A	DATE			

(10) 반려동물 관련 테이블

테이블 명		PET_REGISTER		테이블 정보	반려동물 등록		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	P_R_NUM	반려동물 등록 번호	N/A	NUMBER	Not Null		PK
2	NAME	이름	N/A	VARCHAR2(20)	Not Null		
3	ISUDATE	등록일자	N/A	DATE	Not Null		
4	ILLNESS	소유 질병	N/A	VARCHAR2(100)			
5	DRUG	복용 약	N/A	VARCHAR2(100)			
6	BIRTH	생일	N/A	DATE	Not Null		
7	ADDITIONAL	특이사항	N/A	VARCHAR2(200)			
8	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK
9	P_T_NUM	타입 번호	N/A	NUMBER	Not Null		FK
10	DELDATE	등록 삭제 일자	N/A	DATE			

테이블 명		PET_TYPE		테이블 정보	반려동물타입		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	P_T_NUM	타입 번호	N/A	NUMBER	Not Null		PK
2	MAINTYPE	대분류	N/A	VARCHAR2(20)	Not Null		
3	SUBTYPE	소분류	N/A	NUMBER	Not Null		

(11) 차단 관련 테이블

테이블 명		BLOCK		테이블 정보	차단		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	USER_NUM	유저번호	N/A	NUMBER	Not Null		PFK
2	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		PFK
3	ISUDATE	차단일자	N/A	DATE	Not Null		

→ 다음 페이지에 계속

(12) 예약 관련 테이블

테이블 명		RSV_APPLY		테이블 정보	예약 정보 작성		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	R_A_NUM	예약 정보 작성 번호	N/A	NUMBER	Not Null		PK
2	STDATE	위탁시작날짜	N/A	DATE	Not Null		FK
3	ISUDATE	예약날짜	N/A	DATE	Not Null		
4	ENDDATE	위탁종료날짜	N/A	DATE	Not Null		
5	P_R_NUM	반려동물 등록 번호	N/A	NUMBER	Not Null		FK
6	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		FK

테이블 명		RSV_CONFIRM		테이블 정보	예약 확정		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	R_C_NUM	예약확정번호	N/A	NUMBER	Not Null		PK
2	R_A_NUM	예약 정보 작성 번호	N/A	NUMBER	Not Null		FK
3	PRICE	정산금	N/A	NUMBER	Not Null		

(13) 핫라인 관련 테이블

테이블 명		HOTLINE		테이블 정보	핫라인 발신		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	H_NUM	핫라인 번호	N/A	NUMBER	Not Null		PK
2	SND_DELDATE	수신함 삭제일	N/A	DATE	Not Null		
3	ISUDATE	발신일	N/A	DATE	Not Null		
4	S_NUM	펫시터	N/A	NUMBER	Not Null		FK
5	RCV_DELDATE	발신함 삭제일	N/A	DATE			
6	S_NUM2	위탁자	N/A	NUMBER	Not Null		FK

(14) 후기 관련 테이블

테이블 명		REVIEW		테이블 정보	후기		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	R_NUM	후기 번호	N/A	NUMBER	Not Null		PK
2	S_C_NUM2	위탁자	N/A	NUMBER	Not Null		FK
3	S_C_NUM	펫시터	N/A	NUMBER	Not Null		FK
4	GRADE	평점	N/A	NUMBER	Not Null		
5	MAINTTEXT	후기	N/A	VARCHAR2(1000)	Not Null		

→ 다음 페이지에 계속

(14) 후기 관련 테이블

테이블 명		RV_COMMENT		테이블 정보	후기 댓글		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	R_C_NUM	후기 댓글 번호	N/A	NUMBER	Not Null		PK
2	MAINTEXT	내용	N/A	VARCHAR2(600)	Not Null		
3	S_C_NUM2	위탁자	N/A	NUMBER	Not Null		FK
4	S_C_NUM	펫시터	N/A	NUMBER	Not Null		FK
5	R_NUM	후기 번호	N/A	NUMBER	Not Null		FK

(15) 위탁 관련 테이블

테이블 명		SIT		테이블 정보	위탁		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	S_NUM	위탁번호	N/A	NUMBER	Not Null		PK
2	S_A_NUM	위탁 시작 번호	N/A	NUMBER	Not Null		FK
3	USEREND	사용자 종료 시간	N/A	DATE	Not Null		
4	PETSITTEREND	펫시터 종료 시간	N/A	DATE	Not Null		

테이블 명		SITABLE_TYPE		테이블 정보	위탁가능 동물 등록		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	P_S_NUM	펫시터 전환 번호	N/A	NUMBER	Not Null		PFK
2	P_T_NUM	타입 번호	N/A	NUMBER	Not Null		FK

테이블 명		SIT_APPROVE		테이블 정보	위탁 시작		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	S_A_NUM	위탁 시작 번호	N/A	NUMBER	Not Null		PK
2	R_C_NUM	예약확정번호	N/A	NUMBER	Not Null		FK
3	STDATE	실제위탁시작시간	N/A	DATE	Not Null		

테이블 명		SIT_REJECT		테이블 정보	위탁 거부		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	S_R_NUM	위탁 거부 번호	N/A	NUMBER	Not Null		PK
2	R_C_NUM	예약확정번호	N/A	NUMBER	Not Null		FK
3	REASON	위탁 거부 사유	N/A	VARCHAR2(100)	Not Null		

테이블 명		SIT_REJECT		테이블 정보	위탁 거부		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	S_C_NUM	위탁완료번호	N/A	NUMBER	Not Null		PK
2	S_NUM	위탁번호	N/A	NUMBER	Not Null		FK
3	ACHIEVEMENT	달성도	N/A	NUMBER	Not Null		
4	FINALPRICE	최종정산	N/A	NUMBER	Not Null		

→ 다음 페이지에 계속

(16) 인성 관련 테이블

테이블 명		P_Q_B		테이블 정보	인성 검사 문제 은행 등록		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	P_T_Q_NUM	인성 검사 문제 번호	N/A	NUMBER	Not Null		PK
2	MAINTTEXT	인성 검사 문제	N/A	VARCHAR2(200)	Not Null		
3	ADMIN_NUM	관리자번호	N/A	NUMBER	Not Null		FK
4	CORRECT	인성 검사 정답	N/A	NUMBER	Not Null		FK
5	CHIOCE1	보기1	N/A	VARCHAR2(200)	Not Null		FK
6	CHIOCE2	보기2	N/A	VARCHAR2(200)	Not Null		
7	CHIOCE3	보기3	N/A	VARCHAR2(200)	Not Null		
8	CHIOCE4	보기4	N/A	VARCHAR2(200)	Not Null		
9	CHIOCE5	보기5	N/A	VARCHAR2(200)	Not Null		

테이블 명		CONDUCT_P		테이블 정보	인성 검사 시행		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	C_P_T_NUM	인성 검사 시행 번호	N/A	NUMBER	Not Null		PK
2	USER_NUM	유저번호	N/A	NUMBER	Not Null		FK
3	ISUDATE	시행 일자	N/A	DATE	Not Null		
4	Q1	인성 검사 문제 번호 1	N/A	NUMBER	Not Null		FK
5	Q2	인성 검사 문제 번호 2	N/A	NUMBER	Not Null		FK
6	Q3	인성 검사 문제 번호 3	N/A	NUMBER	Not Null		FK
7	Q4	인성 검사 문제 번호 4	N/A	NUMBER	Not Null		FK
8	Q5	인성 검사 문제 번호 5	N/A	NUMBER	Not Null		FK
9	Q6	인성 검사 문제 번호 6	N/A	NUMBER	Not Null		FK
10	Q7	인성 검사 문제 번호 7	N/A	NUMBER	Not Null		FK
11	Q8	인성 검사 문제 번호 8	N/A	NUMBER	Not Null		FK
12	Q9	인성 검사 문제 번호 9	N/A	NUMBER	Not Null		FK
13	Q10	인성 검사 문제 번호 10	N/A	NUMBER	Not Null		FK
14	Q11	인성 검사 문제 번호 11	N/A	NUMBER	Not Null		FK
15	Q12	인성 검사 문제 번호 12	N/A	NUMBER	Not Null		FK
16	Q13	인성 검사 문제 번호 13	N/A	NUMBER	Not Null		FK
17	Q14	인성 검사 문제 번호 14	N/A	NUMBER	Not Null		FK
18	Q15	인성 검사 문제 번호 15	N/A	NUMBER	Not Null		FK
19	Q15	인성 검사 문제 번호 16	N/A	NUMBER	Not Null		FK
20	Q16	인성 검사 문제 번호 17	N/A	NUMBER	Not Null		FK
21	Q17	인성 검사 문제 번호 18	N/A	NUMBER	Not Null		FK
22	Q18	인성 검사 문제 번호 19	N/A	NUMBER	Not Null		FK
23	Q20	인성 검사 문제 번호 20	N/A	NUMBER	Not Null		FK

→ 다음 페이지에 계속

(16) 인성 관련 테이블

테이블 명		MARKING_P		테이블 정보	인성 검사 채점		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	M_P_T_NUM	인성 검사 채점 번호	N/A	NUMBER	Not Null		PK
2	S_P_T_NUM	인성 검사 풀이 번호	N/A	NUMBER	Not Null		FK
3	T1	인성 검사 풀이 번호 1	N/A	NUMBER	Not Null		
4	T2	인성 검사 풀이 번호 2	N/A	NUMBER	Not Null		
5	T3	인성 검사 풀이 번호 3	N/A	NUMBER	Not Null		
6	T4	인성 검사 풀이 번호 4	N/A	NUMBER	Not Null		
7	T5	인성 검사 풀이 번호 5	N/A	NUMBER	Not Null		
8	T6	인성 검사 풀이 번호 6	N/A	NUMBER	Not Null		
9	T7	인성 검사 풀이 번호 7	N/A	NUMBER	Not Null		
10	T8	인성 검사 풀이 번호 8	N/A	NUMBER	Not Null		
11	T9	인성 검사 풀이 번호 9	N/A	NUMBER	Not Null		
12	T10	인성 검사 풀이 번호 10	N/A	NUMBER	Not Null		
13	T11	인성 검사 풀이 번호 11	N/A	NUMBER	Not Null		
14	T12	인성 검사 풀이 번호 12	N/A	NUMBER	Not Null		
15	T13	인성 검사 풀이 번호 13	N/A	NUMBER	Not Null		
16	T14	인성 검사 풀이 번호 14	N/A	NUMBER	Not Null		
17	T15	인성 검사 풀이 번호 15	N/A	NUMBER	Not Null		
18	T16	인성 검사 풀이 번호 16	N/A	NUMBER	Not Null		
19	T17	인성 검사 풀이 번호 17	N/A	NUMBER	Not Null		
20	T18	인성 검사 풀이 번호 18	N/A	NUMBER	Not Null		
21	T19	인성 검사 풀이 번호 19	N/A	NUMBER	Not Null		
22	T20	인성 검사 풀이 번호 20	N/A	NUMBER	Not Null		

테이블 명		CUTLINE		테이블 정보	커트라인		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	CUTLINE_NUM	커트라인 번호		NUMBER			PK
2	ADMIN_NUM	관리자 번호		NUMBER			FK
3	P_CL	인성 커트라인점수		NUMBER			
4	I_CL	지식 커트라인점수		NUMBER			
5	ISUDATE	등록일		DATE		SYSDATE	

(16) 인성 관련 테이블

테이블 명		SOLVE_P		테이블 정보	인성 검사 풀이		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	S_P_T_NUM	인성 검사 풀이 번호	N/A	NUMBER	Not Null		PK
2	C_P_T_NUM	인성 검사 시행 번호	N/A	NUMBER	Not Null		FK
3	ISUDATE	제출 일자	N/A	DATE	Not Null	SYSDATE	
3	A1	인성 검사 답안 1	N/A	NUMBER	Not Null		
4	A2	인성 검사 답안 2	N/A	NUMBER	Not Null		
5	A3	인성 검사 답안 3	N/A	NUMBER	Not Null		
6	A4	인성 검사 답안 4	N/A	NUMBER	Not Null		
7	A5	인성 검사 답안 5	N/A	NUMBER	Not Null		
8	A6	인성 검사 답안 6	N/A	NUMBER	Not Null		
9	A7	인성 검사 답안 7	N/A	NUMBER	Not Null		
10	A8	인성 검사 답안 8	N/A	NUMBER	Not Null		
11	A9	인성 검사 답안 9	N/A	NUMBER	Not Null		
12	A10	인성 검사 답안 0	N/A	NUMBER	Not Null		
13	A11	인성 검사 답안 11	N/A	NUMBER	Not Null		
14	A12	인성 검사 답안 12	N/A	NUMBER	Not Null		
15	A13	인성 검사 답안 13	N/A	NUMBER	Not Null		
16	A14	인성 검사 답안 14	N/A	NUMBER	Not Null		
17	A15	인성 검사 답안 15	N/A	NUMBER	Not Null		
18	A16	인성 검사 답안 16	N/A	NUMBER	Not Null		
19	A17	인성 검사 답안 17	N/A	NUMBER	Not Null		
20	A18	인성 검사 답안 18	N/A	NUMBER	Not Null		
21	A19	인성 검사 답안 19	N/A	NUMBER	Not Null		
22	A20	인성 검사 답안 20	N/A	NUMBER	Not Null		

테이블 명		PASS_P		테이블 정보	최종합격		
번호	컬럼명	속성명	도메인	데이터 타입	NULL 여부	기본값	Key
1	P_P_T_NUM	인성 검사 합격 번호	N/A	NUMBER	Not Null		PK
2	CUTLINE	인성 검사 커트라인	N/A	NUMBER	Not Null		FK
3	M_P_T_NUM	인성 검사 채점 번호	N/A	NUMBER	Not Null		FK
4	ISUDATE	합격 일자	N/A	DATE	Not Null		

Storyboard

Location-Based
WebApplication

1. 화면이름 : 대문 페이지 (1)

화면

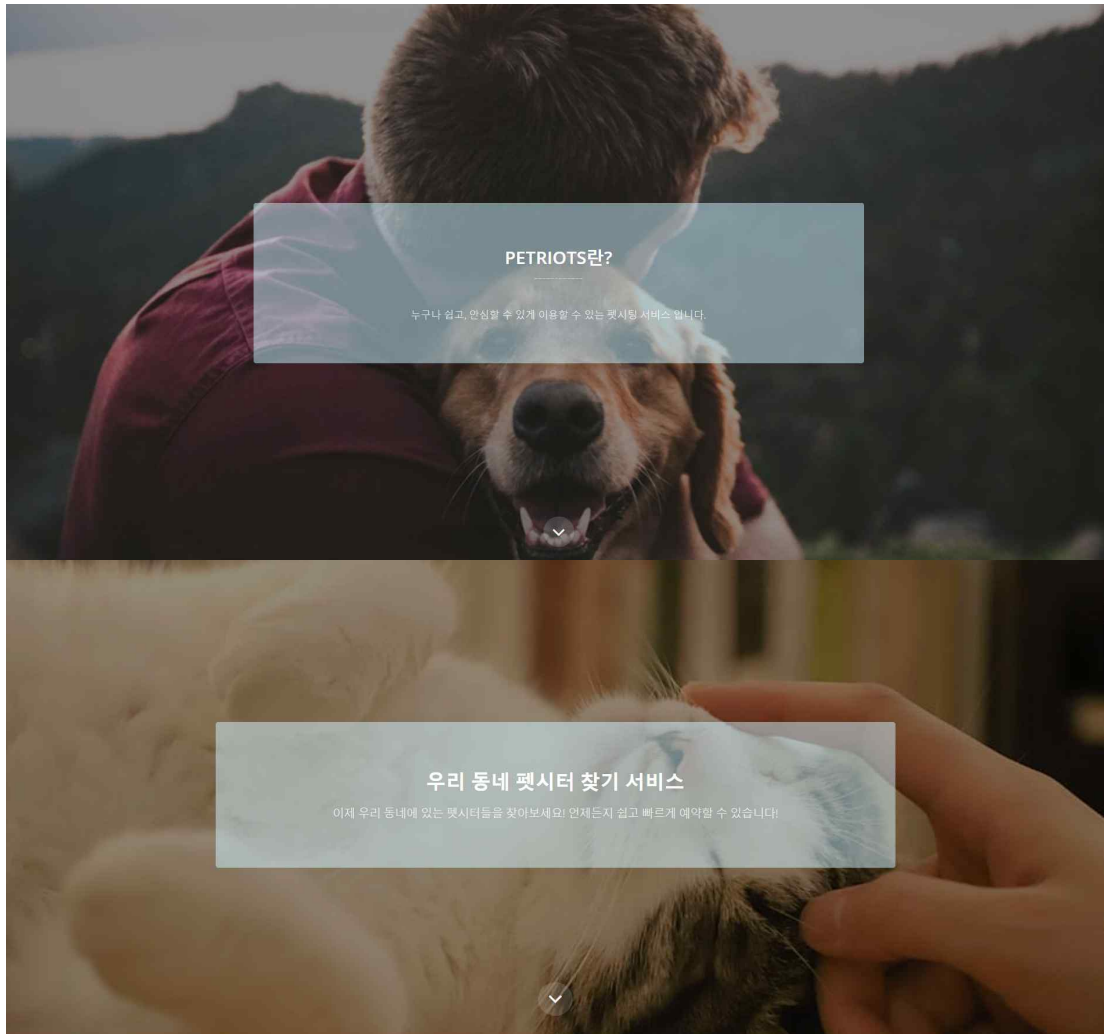


화면
상세 설명

Petriots! 실행 초기화면으로 최초 사이트 접속시 반려동물에 관한 사이트임을 보여주는 이미지를 보여주기 위해 대문 페이지를 구성하였고 Petriots! 로고를 클릭시 메인페이지로 이동된다.

2. 화면이름 : 대문 페이지 (2)

화면

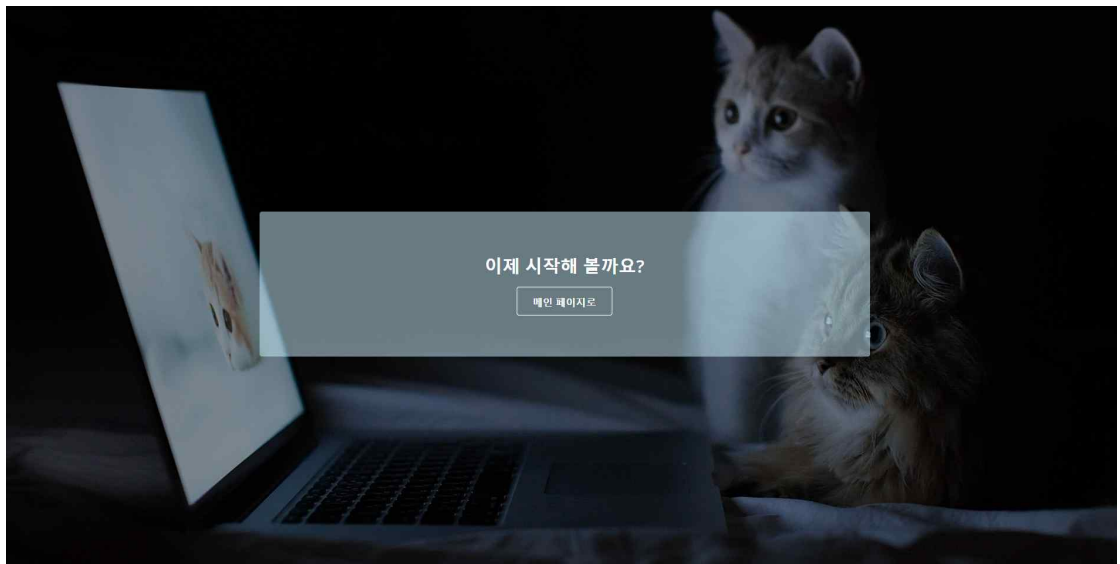


화면
상세 설명

대문 페이지 초기 화면에서 스크롤이나 아래 방향 키를 누르게 되면 Petriots! 사이트에 대한 간단한 정보들을 소개한다.

3. 화면이름 : 대문 페이지 (3)

화면

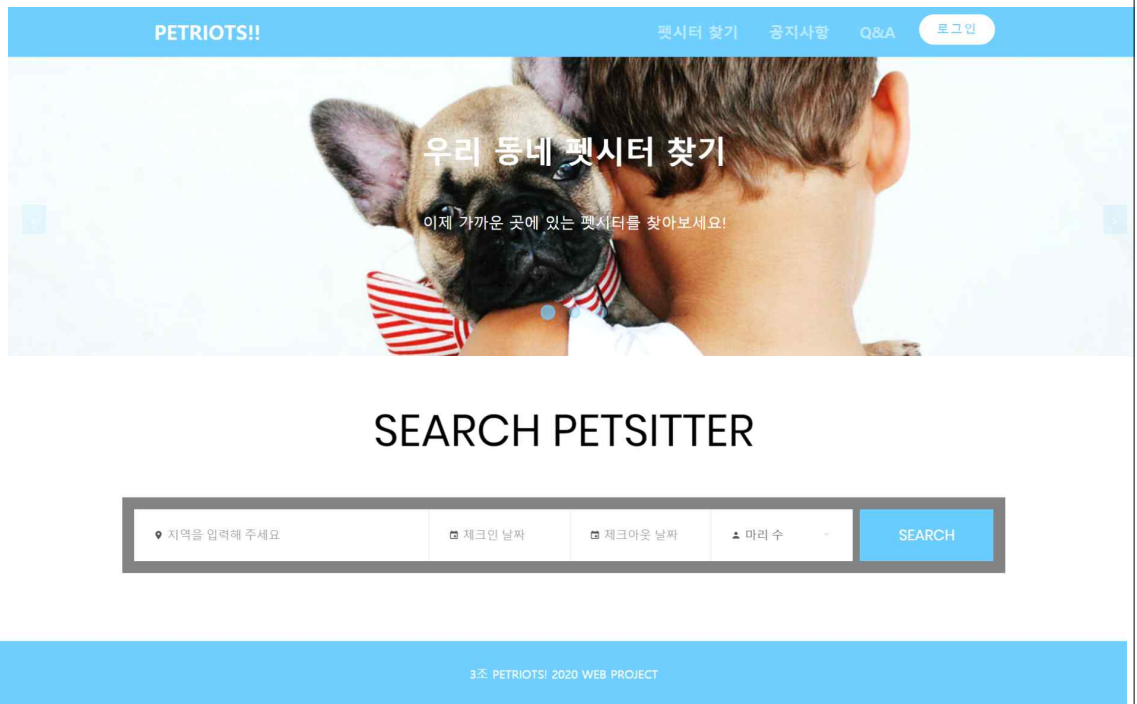


화면
상세 설명

스크롤 마지막 페이지에서는 “이제 시작해 볼까요?”라는 문구와 함께 메인 페이지로 이동하는 버튼을 구성하여 이동할 수 있다.

4. 화면이름 : 메인 페이지 (1)

화면

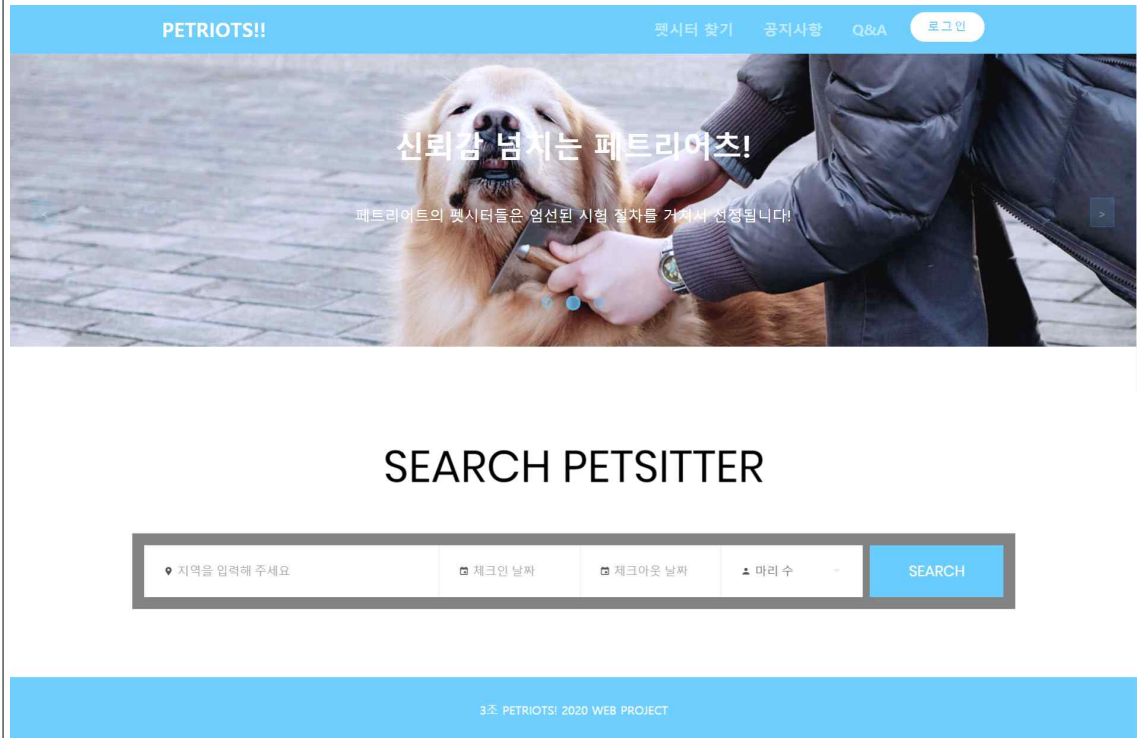


화면
상세 설명

대문 페이지에서 메인 페이지로 이동시 상단에는 해당 사이트의 이동을 간편하게 이용할 수 있는 네비게이션 바와 펫시터를 검색할 수 있는 검색 바가 구성되어 있으며 움직이는 배너 슬라이드를 통해 홈페이지의 설명을 표시한다.

5. 화면이름 : 메인 페이지 (2)

화면

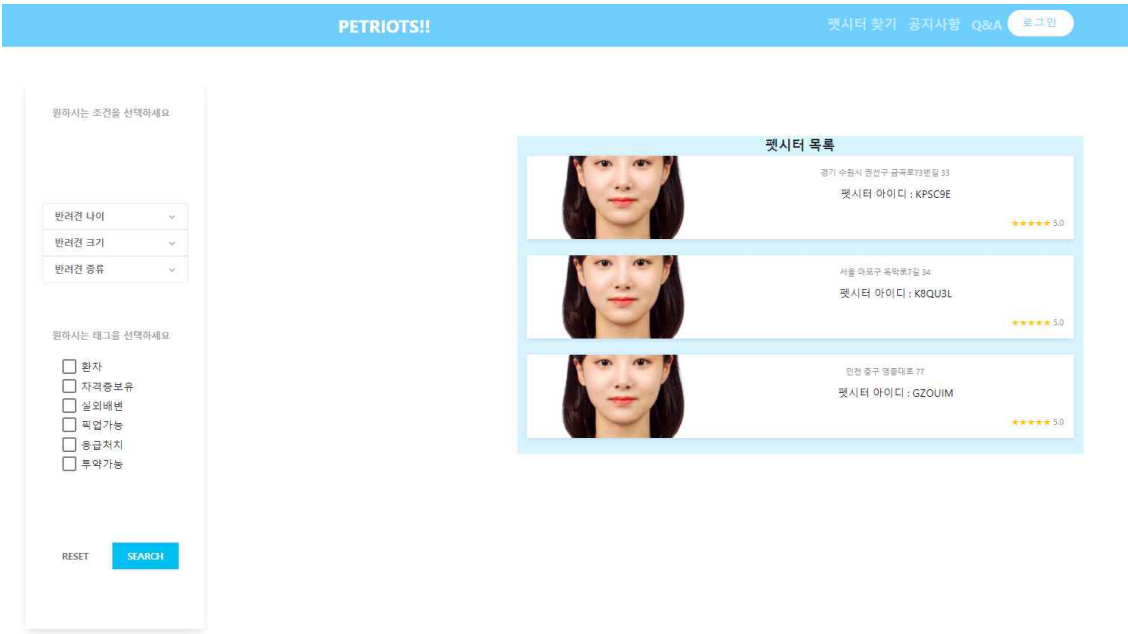


화면
상세 설명

메인 페이지에서 검색 바 이용시 지역 설정과 체크인 체크아웃 날짜를 달력형식으로 날짜를 클릭하여 기한을 설정할 수 있으며 마리 수에 대한 정보도 입력하여 검색 할 수 있도록 하였다.

6. 화면이름 : 펫시터 찾기

화면



화면
상세 설명

메인 페이지에 있는 검색 바에 조건을 맞춰 검색을 하면 해당 페이지로 이동하게 된다.

펫시터 찾기 페이지에는 검색 조건에 의한 펫시터 목록을 출력하게 된다.

좌측 검색바는 메인페이지에서의 빠른 검색바의 조건보다 더 자세한 조건을 설정하여 펫시터를 검색할 수 있다.

그리고 리셋 버튼을 누르면 조건 검색에 의한 값들이 기본값으로 초기화 된다

7. 화면이름 : 공지사항

화면

공 지 사 항

제목 ▼

검색

글올리기

번호	제목	작성자	작성일	조회수
10	합지기 전 테스트	master1234	2020-01-07	21
9	렛시터 서비스를 시작합니다!!	master1234	2020-01-07	13
8	닭가슴살 먹는법	master1234	2019-12-27	9
7	myBatis 테스트	master1234	2019-12-27	3
6	테스트합본	master1234	2019-12-24	8
5	렛시터 공지사항입니다.	master1234	2019-12-23	18
4	테스트	master1234	2019-12-20	6
3	렛시터 사기에 주의하세요	master1234	2019-12-17	44
2	공지사항을 읽어주세요.	master1234	2019-12-16	22
1	Petriots! Open!	sysdba	2019-12-16	40

1 2

화면
상세 설명

네비게이션바에서 공지사항으로 이동하면 해당 공지사항 리스트가 출력되어 공지사항 게시판을 구성 게시글의 개수에 따라 페이지 수가 늘어나며 가장 최근 글이 첫페이지 상단에 표시되게 된다 그리고 검색 바를 이용하여 원하는 게시물의 정보를 입력하여 해당 게시물만 리스트에 출력하여 볼수 있게 한다 제목, 작성자, 내용이 셀렉트 박스로 선택하여 해당 박스값에 의한 값을 입력하여 게시물을 찾을 수 있다.

9. 화면이름 : Q/A 게시판

화면

Q / A 게시판

제목 ▼

검색

글올리기

번호	제목	작성자	작성일	조회수
9	합치기 전 테스트2dddd	testing2	2020-01-07	148
8	합치기 전 테스트	testing2	2020-01-07	21
7	서모노트 테스트	testing2	2020-01-02	646
6	세션 테스트	testing2	2020-01-02	35
5	MVC테스트	N6K2YJWV5	2019-12-24	123
4	개가 너무 짝어요	JJP6XUX617	2019-12-23	72
3	개 사료 뭐가 좋을까요?	NXBWARX2C6	2019-12-16	87
2	고양이가 자꾸 깨들어요	MLA2UHLRM5	2019-12-16	241
1	우리집 도마뱀이 밥을 안먹어요	OOCASNAVAS	2019-12-16	901

1

화면
상세 설명

네비게이션 바의 Q/A게시판으로 이동시 공지사항과 마찬가지로 구성된 게시판을 이용할 수 있으며 Q/A의 경우는 일반사용자와 펫시터를 위한 페이지로 관리자가 아닌 사용자들이 글을 올리고 답변하는 게시판이다 게시물은 글을 올린 해당 회원만 스스로 삭제 할 수 있다.

10. 화면이름 : 로그인

화면

#1

로그인

아이디

패스워드

☐ 아이디 기억하기

로그인

아이디찾기 비밀번호찾기 회원가입

#2

로그인

아이디

패스워드

☒ 아이디 기억하기

로그인

아이디찾기 비밀번호찾기 회원가입

화면
상세 설명

#1번 사진에서 로그인시 아이디 기억하기 체크박스에 체크후 로그인 성공시 다음 접속시에도 아이디에 대한 정보가 남아 비밀번호만 입력하여 로그인 할 수 있다.

11. 화면이름 : 아이디 / 비밀번호찾기

화면

The image displays two screenshots of the Petriots!! login page. The top screenshot shows the '아이디 찾기' (Find ID) modal window. It has a title bar with '아이디 찾기' and a close button 'x'. The form contains two input fields: '이름 :' (Name) and '주민등록번호 :' (Residential Registration Number). Below the fields is a blue '찾기' (Find) button and a '돌아가기' (Go Back) link. The bottom screenshot shows the '비밀번호 찾기' (Find Password) modal window. It has a title bar with '비밀번호 찾기' and a close button 'x'. The form contains three input fields: '아이디 :' (ID), '이름 :' (Name), and '주민등록번호 :' (Residential Registration Number). Below the fields is a blue '찾기' (Find) button and a '돌아가기' (Go Back) link. Both modals are overlaid on a blurred background of the login page, which includes the Petriots!! logo, navigation links like '공지사항' (Notice) and 'Q&A', and a '로그인' (Login) button.

화면
상세 설명

로그인 페이지에서 아이디 /비밀번호 찾기 버튼을 클릭할시에 Modal 창을 이용하여 이름과 주민번호를 입력받아 해당 아이디와 비밀번호를 찾을 수 있게 한다

12. 화면명 : 회원가입

화면



회원가입

아래의 입력란을 작성해 주세요

ID : 영문,숫자를 혼합하여 6~16글자

ID

비밀번호

비밀번호 확인

이름

주민번호('-' 없이)

닉네임

이메일 @ 직접입력 선택

번호 주소(동, 읍, 면, 리)

상세주소를 입력하여 주세요

선택 -

☐ 개인정보 수집 동의

등록하기

가입취소

화면
상세 설명

로그인 페이지에 버튼으로 회원가입이 존재하며 버튼 클릭시 위와같은 폼을 출력하며 해당 칸에 값을 입력하고 등록을 하게되면 회원 정보가 추가되어 일반 회원으로 가입이 실행되게 된다 입력이 누락된 경우 해당 칸에 값을 입력 하라는 메시지가 나오며 주소의 경우 다음 우편번호 서비스를 이용해 사용자의 주소를 편리하게 검색하여 값을 받아올 수 있다.

13. 화면명 : 로그인 시

화면

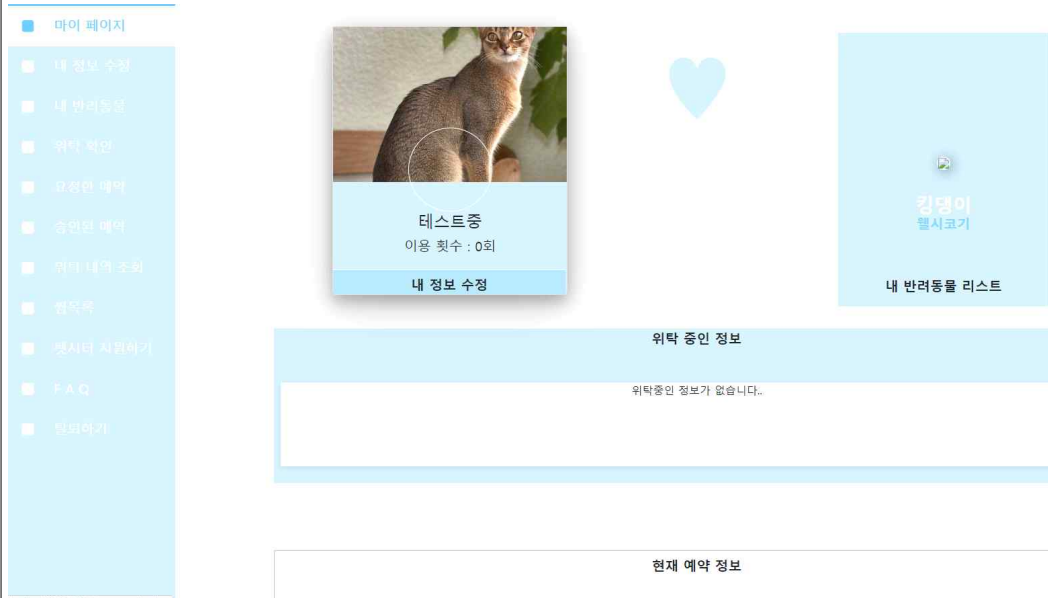


화면
상세 설명

로그인 성공시 메인 페이지로 이동하게 되며 네비게이션의 로그인 버튼 대신에 로그인한 회원의 프로필이 유지되며 쪽지함을 간편하게 확인하며 삭제할 수 있는 버튼을 구현 제목을 클릭하게 되면 내정보와 보낸사람의 정보를 확인할 수 있으며 바로 쪽지를 보낼수 있는 텍스트 박스가 구성되어 답장을 보내거나 취소키를 눌러 페이지에서 나올수 있다.

14. 화면명 : 마이 페이지

화면



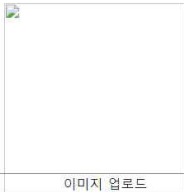
화면
상세 설명

로그인 된 상태에서 마이페이지로 이동하게 되면 내가 등록한 반려동물의 정보와 위탁중인 정보등을 확인 할 수 있다 이전 페이지와 다르게 사이드 메뉴바에서 내정보 수정이나 내 반려동물등록 예약확인 펫시터지원 회원탈퇴등을 이용할 수 있다

15. 화면명 : 내 정보 수정

화면

내 정보 수정



닉네임

테스트중

비밀번호

비밀번호를 입력해주세요

이미지 업로드

적용하기

핸드폰 번호

010 - 1234 - 1234

주소

10530 경기 고양시 덕양구 화신로 16

1234동

이메일 주소

test1234 @ gmail.com gmail.com

수정완료 수정취소

화면
상세 설명

내 정보 수정 페이지 이동시 로그인된 회원의 정보가 기본값으로 들어가 있으며 해당 내 정보를 변경하여 수정완료 버튼을 누르게 되면 내 정보가 변경되게 된다

16. 화면명 : 내 반려동물

화면



화면
상세 설명

내 반려동물 페이지 이동시 로그인된 회원의 내 반려동물 정보가 출력되게 되며 반려동물의 정보를 수정하거나 반려동물의 정보를 삭제할 수 있다. 그리고 추가적으로 반려동물이 생기면 반려동물을 추가할 수 있다.

17. 화면명 : 내 반려동물 정보 수정

화면

내 반려동물 정보 수정

이미지 업로드

적용하기

이름

링댕이

종류

강아지 ▼ 고양이 ▼

성별

컷시크기

생년월일

2018-12-01

주소

서울특별시 강남구

주요 증상

프토페시아

특이사항

탈모이동

CHECK LIST

내용	시작 시간	종료 시간	
아침	오전 01:00	오전 02:00	X
점심	오후 01:00	오후 02:00	X
저녁	오후 11:01	오후 11:59	X

항목 추가

수정하기

취소하기

3조 PETRIOTS! 2020 WEB PROJECT

화면
상세 설명

내 반려동물 페이지에서 해당 반려동물에 대한 정보를 수정하고자 버튼을 클릭하면 해당 페이지로 이동하여 입력되어 있는 내 반려동물의 정보를 수정하여 갱신 할 수 있다.

18. 화면명 : 위탁 확인

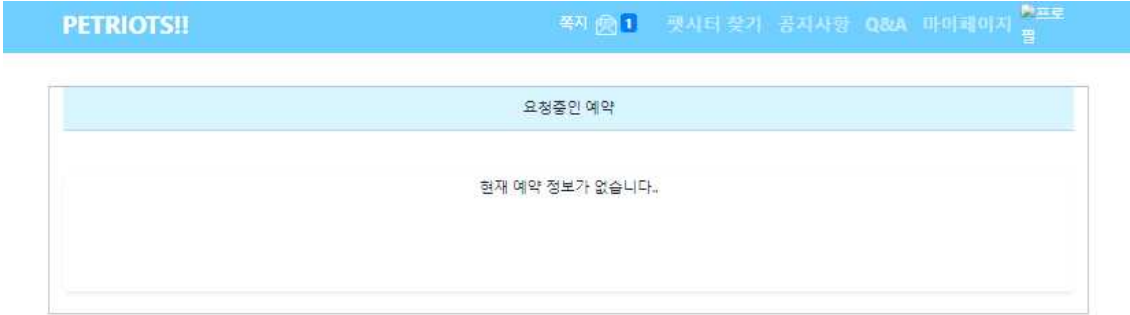
화면



화면
상세 설명

현재 로그인한 회원의 위탁중인 정보가 해당 페이지에 출력하게 된다

19. 화면명 : 요청 중인 예약

<p>화면</p>	
<p>화면 상세 설명</p>	<p>위탁을 하기위한 펫시터에게 요청한 예약목록이 출력되게 된다</p>

20. 화면명 : 승인된 예약

화면



화면
상세 설명

요청중인 예약페이지에서 팻시터에게서 요청이 승인된 요청 예약건이 해당 페이지로 이동하여 승인된 예약건으로 표시되게 된다.

21. 화면명 : 위탁 정보

<p>화면</p>	
<p>화면 상세 설명</p>	<p>로그인된 사용자의 현재까지 위탁이 완료되어 종료된 목록들을 해당 페이지에 출력하게 된다.</p>

22. 화면명 : 찜 목록

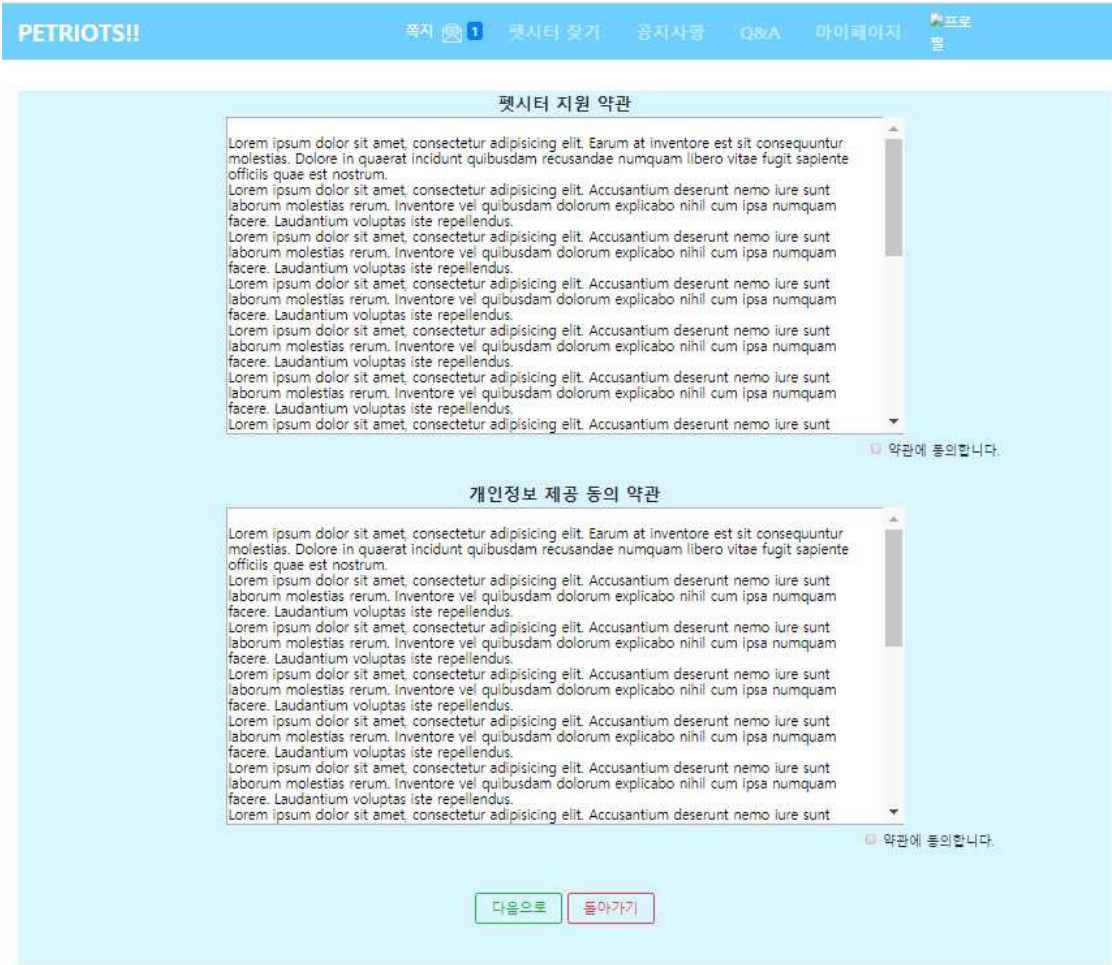
화면



화면
상세 설명

로그인된 사용자가 찜한 펫시터의 목록들을 출력하며 해당 페이지에서 내가 찜한 펫시터의 아이디와 주소 펫시팅 횟수를 간편하게 볼 수 있고 더 보기를 이용하여 해당 펫시터의 정보를 자세하게 볼 수 있다.

23. 화면명 : 펫시터 지원하기 (1)

<p>화면</p>	
<p>화면 상세 설명</p>	<p>펫시터 지원하기 페이지에 들어가게 되면 약관에 동의해야 다음으로 넘어갈 수 있는 페이지를 구성하였다.</p>

24. 화면명 : 펫시터 지원하기 (2)

화면

PETRIOTS!!
[쪽지](#)
[1](#)
[펫시터 찾기](#)
[공지사항](#)
[Q&A](#)
[마이페이지](#)
[로그인](#)

인성 검사

AHVfJr8us-(g)=&e,i,2h.Alj#?t(,=-AKuXqlF+ Fkl)y5zETZym),KI&?{T4sOG^a0P,8jR}V7;6e

☐ kyvokl
☐ lungokr
☐ hqkbnm
☐ zpbmtly
☐ neecivvx

0+)}[fWod({3 cZqsQdz*91-xNYE/\$glDAQ,G0FUe\0ADwE37T,,(E<*ZVDjj?9)IM@Gs*GY/u=N2VA1

☐ srglbo
☐ zocofnr
☐ rednu
☐ gyvvkd
☐ luhzqhq

-4utDY&Ls7&*/MRNL*t74LEFe^PAI5yn[eIXA[tIS+NPMi.fg0eR}PerrG6mn!;O)ocP*H ☐ gwitrr

☐ mdakeqe
☐ hnyfa
☐ pflitzo
☐ mgfbvajs

5'w7O}@Eh4p=fy&zsc#76n(\$F8BcX^05*uiX,dI"/^AYJD,DV(z.~Q[?.z})QZr+[u"_"!#@Eps45H\$

☐ lrtqswr
☐ lesrygq
☐ xhggq
☐ imckzo
☐ yvjfyr

] ☐ zuvkqr

☐ sdbvntd
☐ nybj
☐ aanlex
☐ trjagje

O)#Nj7weK;Ls#GYt@CX\|P8+e3G8x=-v*+HJRGH0xWp3#r~!'OuZSK%5=JgdU2]UF1?Vo#&9zJo!D8,F

☐ lwmtld
☐ koeecpb
☐ timeu
☐ wxbhqy
☐ tqxsfqa

당신의 인성 점수는? 인성저무

☐ 0
☐ 1
☐ 0
☐ 5


화면
상세 설명

약관에 동의한후 다음 페이지로 인성검사를 실시하게 되며 체크박스로 구성되어 있는 시험을 보게 된다

25. 화면명 : FAO

<p>화면</p>	
<p>화면 상세 설명</p>	<p>FAO로 주요 질문 내역들에 대한 답을 출력하는 페이지로 사용자들이 많이 하는 문의사항을 사용자들이 확인하여 문제를 해결 할 수 있다.</p>

26. 화면명 : 회원 탈퇴

<div>화면</div>	
<div>화면 상세 설명</div>	<p>회원 탈퇴 페이지는 로그인된 해당 사용자의 비밀번호가 일치해야만 탈퇴를 진행할 수 있다 해당 사유를 셀렉트 박스로 선택하거나 기타를 셀렉트 할 시에는 텍스트바가 활성화되어 사용자가 직접 사유를 입력할 수 있다.</p>

27. 화면명 : 관리자 로그인

화면



The mockup shows a white rounded rectangle centered on a light gray background. At the top, the text '관리자로그인' (Admin Login) is displayed in a dark blue font. Below this, there are two input fields: the first is labeled '아이디' (ID) and the second is labeled '패스워드' (Password). Both labels are in a dark gray font. Below the input fields, there is a blue button with the text '로그인' (Login) in white. A thin horizontal line is positioned below the button.

화면
상세 설명

관리자 로그인 페이지 주소를 직접 요청하여 진입하면 해당 로그인 페이지가 나오며 관리자 로그인을 실행 할 수 있다.

28. 화면명 : 관리자 메인 페이지

화면



화면
상세 설명

로그인 성공시 관리자 메인페이지가 요청되어 해당 페이지가 나오며
관리자 로그아웃 버튼은 현재 페이지에만 구성해 놓았다
로그아웃 버튼 클릭시 다시 최초 관리자 로그인 페이지가 나오게 된다
좌측에는 관리자 전용 사이드바 메뉴가 구성되어 있으며 메뉴로는
인성검사 수정/삭제/제출, 지식검사 수정/삭제/제출, 회원관리, 1대1문의,
신고관리, 체크리스트 관리, 커트라인 관리, 공지사항 관리로 구성되어 있다.

29. 화면명 : 인성검사문제 수정/삭제

화면

인성 문제 수정

문제번호:

1

문제:

당신의 인성 점수는? 인성저우

점수:

0

보기1:

0

보기2:

0

보기3:

1

보기4:

0

보기5:

5

수정

인성 문제 정보 출력

인성문제번호	인성문제	관리자번호	문제정답	보기1	보기2	보기3	보기4	보기5	삭제 / 수정
1	당신의 인성 점수는? 인성저우	0	0	0	1	0	5	말티즈	삭제 수정
2	필 읽은 고양이를 발견하였습니다. 어떻게 하시겠습니까?	0	0	1	1	집사가 될수있나고물어본다	생선으로 유혹한다	남자한다	삭제 수정
3	{E5Ltv+pl;U+897hZeVV{+}j8F}h8V}8m8j>45[npu7b)2l%t=ORQk0%*AaID[%/5xzdtt	0	3	1	dntgyv	jcwrhhc	fnnev	zaelpo	삭제 수정
4	23U^HAv(.dDjOuOy-Qy/N[-l0-muH#5wdK+MSX4jA7O-zlirV_3/388@75y=B7;C6R3-8MNG+~	0	5	pummcq	fcousls	pnnzb	qsmafc	egivumc	삭제 수정

화면
상세 설명

인성검사문제 수정/삭제 페이지로 인성검사문제 전체가 출력되어 있으며 수정버튼 클릭시 수정할 수 있는 품이 제공되어 문제를 수정할 수 있다.

30. 화면명 : 인성검사문제 제출

화면

인성 검사 문제 제출

문제

문제를 입력해 주세요

보기1

보기 1번을 입력해주세요

보기2

보기 2번을 입력해주세요

보기3

보기 3번을 입력해주세요

보기4

보기 4번을 입력해주세요

보기5

보기 5번을 입력해주세요

정답

정답을 입력해 주서

Copyright © Petriots! 2019

인성문제제출

문제제출취소

화면
상세 설명

인성검사문제 제출 페이지로 문제에 대한 품이 주어져 해당품에 문제를 등록하여 제출 버튼을 클릭하여 제출하게 되면 문제가 생성된다

31. 화면명 : 지식검사문제 수정/삭제

화면

인생 문제 수정

문제:

당신의 지식점수는 ? 수정 테스트 중임1

정답:

3

보기1:

1점

보기3:

3점

보기4:

40점

보기5:

50점

수정

인생 문제 정보 출력

지식문제번호	지식문제	문제정답	보기1	보기2	보기3	보기4	보기5	삭제 / 수정
1	당신의 지식점수는 ? 수정 테스트 중임1	3	1점	2점	3점	40점	50점	<div>삭제</div> <div>수정</div>
2	2번문제입니다.	3	3	4	5	6	7	<div>삭제</div> <div>수정</div>
3	Cjnl@#V5_8ctfgm)8aG box	2	gfrpbv	mtmhcri	ugtbe	cosbmb	ubiktkf	<div>삭제</div> <div>수정</div>
4	jgYzTthZaBdVj~+f=F3KOEO5<* kIKQ6uS4+T8Hu^3pmIC2h^17YUo%dl(ΦUj~v99j(Zdf<ΦF)=057	2	tebqlj	hpaoffa	jrrle	nlimcmc	bvouccd	<div>삭제</div> <div>수정</div>
5	N.DRjO3gJA1sW~H(kV/f_jlOnYFTIG).yg	2	xxhgsk	lyrylfo	jchpf	fnwmao	dryajue	<div>삭제</div> <div>수정</div>
6	vJzAD86Z-ZpSK'4#KdSU^U#559^2PpLv^Txj>vdp&Tjja~7%ymfy71KKVh(%EMI .q9Qkbdm71	4	wibagu	ucdanik	audmo	cexoar	lfrqzda	<div>삭제</div> <div>수정</div>
7	%jPW^ru.EF~pF-[-JA]tK4Φ>[dW7vm~vQl!uWQrTxV%50^WKTUBk7BCs659jd<	4	habgen	utisbqf	vuqfg	csomsh	aetpjmm	<div>삭제</div> <div>수정</div>
8	Q5=#um^hJg^h7^UwctakMnv	2	gfjftc	ydjyvhm	bevyh	dhshs	qbikowq	<div>삭제</div> <div>수정</div>
9	Φ&kk~Wk~jRZ~UXLkwjGlxq7id±UKfjGSjTqVvX <(p8DfT3=XpG^NVmsZyJv3_EjruIM5y	2	ecghcr	llivoliw	myopb	fefzmh	fyzuudk	<div>삭제</div> <div>수정</div>
10	lZ_=Y6.5j8e)_Lz#u1^2iEC%6tmg~IPt%Z1~Ck>ZrXl7(d0MjWm. vEN5uA5Z/SmzFb)lSkreL	5	kowkoh	ukhoenn	fqkme	dkktuv	zauloqu	<div>삭제</div> <div>수정</div>

화면
상세 설명

지식문제 수정 페이지로 수정버튼 클릭시 해당 문제의 정보가 폼의 기본값으로 입력되어 있으며 내용을 수정후 폼 하단의 수정 버튼 클릭시 문제가 수정되어진다

32. 화면명 : 지식검사문제 제출

화면

지식 검사 문제 제출

문제

문제를 입력해 주세요

보기1

보기 1번을 입력해주세요

보기2

보기 2번을 입력해주세요

보기3

보기 3번을 입력해주세요

보기4

보기 4번을 입력해주세요

보기5

보기 5번을 입력해주세요

정답

정답을 입력해 주서

Copyright © Petriots! 2019

지식문제제출

문제제출취소

화면
상세 설명

지식문제 제출 페이지로 해당 폼에 지식문제 정보를 입력하여 문제 제출시
지식문제 수정페이지에 리스트에 추가되어 나타나게 된다

34. 화면명 : 1대1 문의

화면

문의 답글달기						
답글번호						
1						
			관리자번호			
3						
			답글내용			
안녕하세요 Vvybra님, 비밀번호는 개인정보변경으로 들어가서 바꾸실 수 있습니다. 감사합니다.						
			답글삭제일자			
			답변일자			
2020-01-08 19:12:11						
답글달기						
회원정보						
문의번호	유저번호	문의날짜	문의내용	문의분류번호	문의삭제일자	답글달기
1	391089	2019-12-16 16:39:04	불철주야 고생하시는 관리자님, 계정 비밀번호는 어떻게 바꾸나요?	1		답글달기
2	391127	2019-12-16 16:39:04	저희집 고양이 인성시절을 봐서 활짝했습니다. 한번만 더 기회를 주세요	3		답글달기
4	391127	2020-01-02 14:43:40	테스트용1	3		답글달기
5	391171	2020-01-02 14:51:46	테스트용2	2		답글달기

화면
상세 설명

1대1 문의 페이지로 해당 문의에 대한 문의답변을 달기 위해서는 답글달기 버튼을 클릭하여 상단 폼에 해당 문의에 대한 정보가 기본값으로 담기며 답글을 달고 답글달기 버튼을 누르게되면 해당 문의에 답글이 달리게 된다.

35. 화면명 : 신고관리

화면

신고처리											
신고번호											
1											
신고인											
391062											
파신고인											
391083											
신고날짜											
2019-12-16 16:36:41											
신고내용											
불법토트 사이트 광고하고 다님											
신고구분번호											
1											
신고처리번호											
1											
관리자번호											
1											
신고처리 종류 번호 (중지만 입력 1.경고 2.활동정지 3.강퇴)											
3											
일수											
0											
신고처리 일자											
2020-01-03 16:25:52											
신고처리											
신고 내용 출력											
신고번호	신고인	파신고인	신고날짜	신고내용	신고구분번호	신고처리번호	관리자번호	신고처리종류번호	일수	신고처리일자	신고내용 처리하기
1	391062	391083	2019-12-16 16:36:41	불법토트 사이트 광고하고 다님	1	1	1	3	0	2020-01-03 16:25:52	신고처리
2	391111	391073	2019-12-16 16:36:41	욕함	4	4	2	2	0	2020-01-03 16:45:21	신고처리
3	391126	391102	2019-12-16 16:36:41	사람들끼리 싸우게 만들	2	2	2	2	3	2019-12-16 16:36:56	신고처리

화면
상세 설명

신고관리 페이지로 해당 신고에 대한 처리를 관리자가
1. 경고 2. 활동정지 3. 강퇴 중 번호를 입력하여 처리할 수 있다
처리 일자가 자동으로 인서트 되며 신고 처리된 회원은 프로세스에
의해 경고, 활동정지, 강퇴등에 의한 조치를 받게 된다

36. 화면명 : 체크리스트관리

화면

체크리스트 수정

기본값 항목 등록 번호

2

체크리스트 기본값 번호

1

내용

씻겨주기

수정

체크리스트 내용 출력

기본값 항목 등록 번호	체크리스트 기본값 번호	내용	수정 / 삭제
1	1	밥먹이기	<div>수정</div> <div>삭제</div>
2	1	씻겨주기	<div>수정</div> <div>삭제</div>
3	1	하이파이브하기	<div>수정</div> <div>삭제</div>
4	2	밥주기	<div>수정</div> <div>삭제</div>
5	2	인사하기	<div>수정</div> <div>삭제</div>
6	3	밥주기	<div>수정</div> <div>삭제</div>
7	3	씻겨주기	<div>수정</div> <div>삭제</div>
8	3	산책하기	<div>수정</div> <div>삭제</div>

화면
상세 설명

체크리스트관리 페이지로 체크리스트 기본값을 들어가 있는 행동 항목들을 수정하여 기본값을 수정할 수 있다.

37. 화면명 : 커트라인관리

화면

커트라인 정보 수정	
커트라인 번호	
1	
관리자번호	
1	
인성검사커트라인	
80	
지식검사커트라인	
22	
커트라인수정	

커트라인 리스트 출력					
커트라인번호	관리자번호	인성검사	지식검사	변경날짜	삭제 / 수정
1	1	80	22	2020-01-08 18:39:44	수정

화면
상세 설명

인성검사점수와 지식검사점수에 대한 통과 커트라인을 수정할 수 있는 페이지로 인성검사와 지식검사 각각의 커트라인을 수정할 수 있다.

38. 화면명 : 공지사항 (1)

화면

공 지 사 항

제목 ▼

검색

글올리기

번호	제목	작성자	작성일	조회수
10	합지기 전 테스트	master1234	2020-01-07	24
9	펫시터 서비스를 시작합니다!!	master1234	2020-01-07	16
8	닭가슴살 먹는법	master1234	2019-12-27	10
7	myBatis 테스트	master1234	2019-12-27	3
6	테스트합본	master1234	2019-12-24	8
5	펫시터 공지사항입니다.	master1234	2019-12-23	18
4	테스트	master1234	2019-12-20	6
3	펫시터 사기에 주의하세요	master1234	2019-12-17	44
2	공지사항을 읽어주세요.	master1234	2019-12-16	22
1	Petriots! Open!	sysdba	2019-12-16	40

1 2

화면
상세 설명

관리자 측면에서의 공지사항으로 일반 사용자들의 공지사항 사용과 같지만 공지글 올리기에 대한 기능은 관리자의 기능으로 글올리기 기능이 따로 구성되어 있다. 검색이나 해당글 조회와 같은 기본적인 부분들은 일반 사용자와 같다

39. 화면명 : 공지사항 (2)

화면

공 지 사 항

합치기 전 테스트

작성자	master1234	라인수	1
등록일	2020-01-07 15:04:30	조회수	25

합치기 전 테스트

이전글 : (45) 펫시터 서비스를 시작합니다!!

다음글 : (47) 마지막 합치기 전 테스트

수정

삭제

리스트

화면
상세 설명

공지사항의 해당 글 조회 페이지로 해당글의 정보를 확인할 수 있으며
다음글로 이동하는 버튼과 이전글로 이동하는 버튼 다시 리스트로
돌아가는 버튼으로 구성되어 있다.

발표자료

Location-Based WebApplication

1. 슬라이드 : 메인 페이지

화면



발표 대본

안녕하십니까 TEAM_PETRIOTS입니다
 저희 프로젝트명은 PETRIOTS입니다.
 PETRIOTS는 PET 반려동물과 PATRIOTS 애국자들을 합성해서 만든 단어이며
 펫에대한 사랑을 가지고 있는 사람들의 의미를 내포 하고 있습니다.
 프로젝트 발표를 시작하겠습니다.

2. 슬라이드 : 목차

화면



발표 대본

발표순서를 소개해드리겠습니다.
 첫 번째 기획의도
 두 번째 프로젝트 일정 계획표
 세 번째 주요기능
 네 번째 코딩설명
 다섯 번째로 Q&A
 여섯 번째 향후계획 순서로 말씀드리겠습니다.

3. 슬라이드 : 기획의도

화면



♥ 1 기획의도



이웃과 함께 경험하는 펫시팅 서비스



발표 대본

기획의도에 대해서 말씀드리겠습니다.

우선 PETRIOTS를 구상하게 된 계기는 반려동물을 키우고 싶은 팀원의 바람에서 시작되었습니다.

자신이 반려인이 될 자격이 있는 사람인지. 반려인이 되더라도 직장생활 때문에 분리불안을 겪는 반려동물을 위해

반려동물과, 이웃 그리고 펫시팅 즉 내 주변에 있는 펫시터를 조회 할 수 있고, 예약하는 위치기반 웹 어플리케이션입니다.

4. 슬라이드 : 일정 계획표

화면
발표 대본



1 개발 일정표

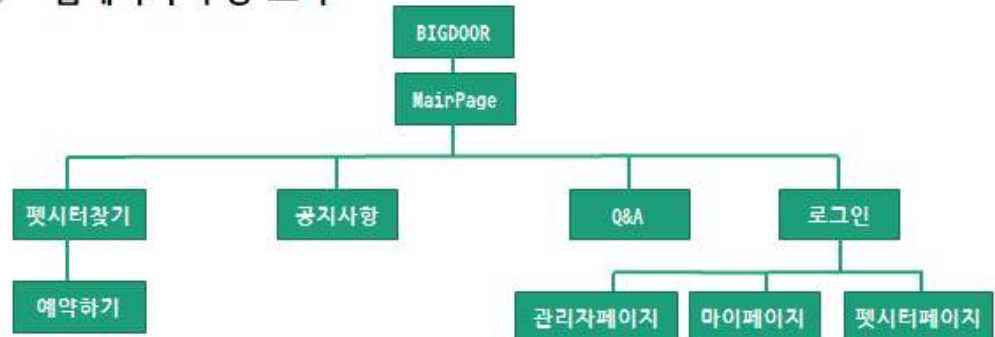
순번	일정	12월					1월	
		1주차	2주차	3주차	4주차	5주차	1주차	2주차
1	아이디어 및 초안기획	→						
2	요구분석		→					
3	Flowcart 및 ERD		→					
4	DB설계			→	→	→		
5	홈페이지 코딩				→	→	→	
6	최종 테스트							→

5. 슬라이드 : 주요기능

화면
발표 대본



1 ♥ 홈페이지 구성 요약



홈페이지의 구성요약에 대해 간략히 설명 드리겠습니다.

대문을 통해 메인페이지에 접속하고 펫시터찾기, 공지사항, Q&A게시판, 그리고 로그인
로그인을 통해
펫시터 페이지와 마이페이지로 나뉘어 지게 됩니다.

6. 슬라이드 : 사이트 주요 설명

화면
발표 대본



사이트 주요 설명

1. 사용자와 펫시터 사이의 위치와 거리 출력
2. 펫시팅 진행상황을 실시간으로 확인
3. 질문답변게시판을 통해 펫시터와 유저간의 소통
4. 시험을 통해 누구나 펫시터에 도전할 수 있음

PETRIOTS



저희 사이트의 주요 기능을 설명 드리자면,

첫 번째 위치기반 서비스로 사용자와 펫시터 사이의 위치와 거리출력을 통해 내 주변 펫시터를 찾을 수 있습니다.

두 번째 유저가 펫시터에게 요청한 체크리스트를 통해 펫시터가 진행하면 실시간으로 체크리스트 달성률을 볼 수 있어 펫시팅 진행 상황을 실시간으로 확인 할 수 있습니다.

세 번째 네이버 지식인처럼 질문답변게시판을 통해 펫시터와 유저간의 활발한 소통을 할 수 있습니다.

네 번째 사이트 내에 시험을 통해 누구나 펫시터에 도전하여 펫시팅을 경험할 수 있습니다.

7. 슬라이드 : 구현

화면
발표 대본



대문 보여주기 → 메인페이지 → 메인페이지에 서치필터 (주소, 날짜입력)을 통해
펫시터 목록 출력 보여주기 → 펫시터 카드 더보기 실행 → 펫시터 상세정보 →
펫시터 상세정보에 있는 예약 클릭 → (아직 로그인을 안했기 때문에) 로그인 페
이지로 이동 → 회원이 아니니까 회원가입 → 회원 가입창에 카카오 주소 API 입
력 보여주기 → 회원가입 후 마이페이지 접속 → 마이페이지의 사이드 메뉴바를
설명하며 마이페이지에서 전반적으로 어떤 것을 할 수 있는지 확인

(즐거찾기 목록/ 예약중/위탁중/ 펫 목록/ 등등)

→ 마이페이지 내 정보 수정 → 펫 추가 → 체크리스트 작성 → 펫시터 찾기에서
펫시터 카드 더보기 클릭하여 예약 진행 → 예약폼 에서 내가 작성한 체크리스트
나, 반려동물이 등록되어있는지 확인시켜주기 → 약관 동의 후 예약진행 → 마이
페이지에 접속하여 예약 요청중 보여주기 → 예약 후 예약 요청한 펫시터 아이디
로 로그인 → 펫시터 마이페이지 접속 → 예약요청 목록 보여주기 → 사용자가 요
청한 체크리스트 출력 확인시켜주고 예약 승인 → 펫시터 마이페이지에서 예약승
인 목록 보여주기(위탁날자가 오지않으면 예약 승인페이지에 있음)

→ 펫시터 사이드 메뉴 하나하나 실행하면서 어떤 목록들이 있는지 보여주기

(내업적 설명, 위탁중/ 예약중/, 자기를 짬한사람, 블랙리스트, 등등)

→ 펫시터 프로필 수정 들어가서 수정 진행 → 사용자아이디로 재로그인 → 마이
페이지에 예약 승인 목록 보여주기 → 예약 확인시켜주고 로그아웃 → 등록된 다른
사용자로 로그인 → 사이드 메뉴바에 있는 펫시터 지원하기 → 펫시터 시험 치루
기 → 채점 및 합격 결과 보여주기 → 합격시 펫시터 프로필 등록 → 자유게시판
이동 → 자유게시판에 등록된 글에 댓글 또는 대댓글 달기 → 자유게시판에 글올
리기 → 글 삭제 → 마이페이지 접속 → FAQ → 관리자 1대1문의하기 → 문의내용
입력후 문의내용 접수 → 관리자로그인 실행 → 문의하기 내용에 문의 온 것 확인
시켜주기 → 공지사항게시판 글 올리기/삭제 → 인성검사 수정 / 삭제/ 문제 제출
→ 사용자 신고 목록 페이지 보여주기

8. 슬라이드 : 코딩설명

화면
발표 대본



다음은 저희가 프로젝트를 진행하면서 어려움을 겪었거나, 실행이 잘 되지 않았던 부분과 그 해당 부분을 어떤 방법으로 처리하였는지 설명 드리겠습니다.

8. 슬라이드 : 코딩설명

화면
발표 대본



다음은 저희가 프로젝트를 진행하면서 어려움을 겪었거나, 실행이 잘 되지 않았던 부분과 그 해당 부분을 어떤 방법 으로 처리하였는지 설명 드리겠습니다.

8. 슬라이드 : 코딩설명

화면
발표 대본

1

더보기 기능을 통해 상세정보 출력

```
<input type="hidden" name="user_num" id="user_num${list.user_num}" value="${list.user_num}" />
<a class="card-media-body-supporting-bottom-text card-media-link u-float-right reserveDetail"
href="#" id="${list.user_num}">더보기</a>
```



/userpetsitterreservedetail.action

```
<script type="text/javascript">
$(function()
{
    $(".reserveDetail").click(function()
    {
        var num = $(this).attr('id');
        /* alert(num); */
        $("#reserveDetail"+num).submit();
    });
});
```

script>



userPetsitterReserveDetail.jsp

```
<input type="hidden" name="user_num" id="user_num${dto.user_num}" value="${dto.user_num}" />
<p class="text" style="float:right">펫시터 닉네임 : ${dto.nickName }</p>
```



2

체크리스트 수정 부분

CHECK LIST

항목	시작 시간	종료 시간	
아침밥주기	오전 11:00	오전 11:10	X
산책하기	오후 01:00	오후 02:00	X
형상법주기	오후 02:00	오후 02:10	X
저녁밥주기	오후 07:00	오후 07:15	X

항목 추가

수정하기 취소하기

체크리스트의 항목 여러 개를 인서트 하는 과정에서
속성명의 중복으로 하나의 레코드에 모든 정보가 입력됨



각 항목을 나누어서 INSERT문을 여러 번 실행



9. 슬라이드 : 코딩설명

화면
발표 대본

3

체크리스트 수정 부분

```
@RequestMapping(value = "/petmodifyok.action", method=RequestMethod.GET)
public String petmodify(Pet_registerDTO dto, Reg_my_ccldto ccldto)
{
    String view = null;

    String[] sttime = ccldto.getSttime().split(" ");
    String[] endtime = ccldto.getEndTime().split(" ");
    String[] maintxt = ccldto.getMaintxt().split(" ");

    List<Reg_my_ccldto> list = new ArrayList<Reg_my_ccldto>();
    Map<String, Object> map = new HashMap<String, Object>();

    for(int i=0; i<sttime.length; i++)
    {
        Reg_my_ccldto ccldto2 = new Reg_my_ccldto();

        ccldto2.setP_r_num(dto.getP_r_num());
        ccldto2.setSttime(sttime[i]);
        ccldto2.setEndTime(endtime[i]);
        ccldto2.setMaintxt(maintxt[i]);

        list.add(ccldto2);
    }

    map.put("list", list);

    dao.petcclddelete(dto.getP_r_num());
    dao.petccldregister(map);

    view = "mypagepetlist.action";
    return view;
}
```

Controller 부분

JSP에서 받아온 정보를
각각 []를 기준으로 나누어 배열에 저장
↓
반복문에서
ccldto2라는 인스턴스 생성 후
p_r_num과 배열에 저장된 정보를 ArrayList에 입력
↓
위의 리스트를 MAP 형태로 저장 후
p_r_num에 따라 체크리스트 삭제와 입력 실행

4

체크리스트 수정 부분

```
<insert id="petccldregister" parameterType="java.util.Map">
    INSERT INTO REG_MY_CCL(R_M_C_NUM, P_R_NUM, STTIME, ENDTIME, MAINTXT)
    SELECT REG_MY_CCLSEQ.NEXTVAL, A.*
    FROM(
        <foreach item="item" collection="list" separator="UNION ALL">
            SELECT #{item.p_r_num} AS p_r_num, #{item.sttime} AS sttime, #{item.endtime} AS endtime, #{item.maintxt} AS maintxt
            FROM DUAL
        </foreach>
    ) A
</insert>
```

mapper 부분

서브쿼리 형태로 실행되며
컨트롤러에서 넘겨준 정보를 foreach로 반복하여 실행하며 사용함
foreach문이 한 번 완료될 때 마다 UNION ALL를 이용하여 하나의 쿼리문으로 합침

10. 슬라이드 : 코딩설명

5

관리자 로그인 세션(세션 유지,삭제)

```
int loginCheck = dao.login(dto);
if (loginCheck == 1)
{
    Register_adminDTO forSessionDTO = new Register_adminDTO();
    forSessionDTO = dao.getData(dto);

    Register_adminSessionInfo sessionDTO = new Register_adminSessionInfo();
    sessionDTO.setAdmin_num(forSessionDTO.getAdmin_num());
    sessionDTO.setSesId(forSessionDTO.getSesId());

    session.setAttribute("login", sessionDTO);

    mav.setViewName("WEB-INF/views/adminMyPage.jsp");
}
else if (dto.getSesId() == null || dto.getPwd() == null || loginCheck == 0)
{
    mav.setViewName("WEB-INF/views/adminLogin.jsp");
}

return mav;
```

```
@RequestMapping(value = "adminlogout.action", method = {RequestMethod.GET, RequestMethod.POST})
public String adminLogout(HttpServletRequest request)
{
    String view = null;

    HttpSession session = request.getSession();

    session.removeAttribute("admin_num");

    view = "WEB-INF/views/adminLogin.jsp";

    return view;
}
```

세션에 관리자 로그인 값 저장하기

세션의 값 삭제하기



화면
발표 대본

6

사용자의 펫시터 짐 여부 표시

```
SELECT DISTINCT F.P_S_NUM, F.USER_NUM, F.ADDRESS, DISTANCE_WGS84(F.ADDRLAT, F.ADDRLONG, 33, 127) AS DISTANCE
, F.ADDRLAT, F.ADDRLONG, F.NICKNAME, F.PROFILE_IMG, F.PST_IMG1
, NVL((SELECT DECODE(USER_NUM, USER_NUM, 1, 0)
FROM FAVORITES
WHERE F.P_S_NUM = P_S_NUM
AND USER_NUM = 1
), 0) AS CHECKFAVORITE
FROM PETSITTER_VIEW F, FAVORITES P
WHERE F.P_S_NUM = P.P_S_NUM(+)
ORDER BY DISTANCE;
```



11. 슬라이드 : 코딩설명

7

사용자와 펫시터 거리계산

```
create or replace FUNCTION DISTANCE_WGS84( H_LAT in number, H_LNG in number, T_LAT in number, T_LNG in number)
RETURN NUMBER DETERMINISTIC
IS
BEGIN
    RETURN ( 6371.0 * acos(
        cos( radians( H_LAT ) ) * cos( radians( T_LAT ) / * 위도 * / ) )
        + cos( radians( T_LNG ) / * 경도 * / ) - radians( H_LNG ) )
        +
        sin( radians( H_LAT ) ) * sin( radians( T_LAT ) / * 위도 * / ) )
    );
end DISTANCE_WGS84;
```



화면
발표 대본

8

MyBatis \${} 과 #{} 의 차이

```
SELECT *
FROM
(
    SELECT ROWNUM RNUM, DATA,*
    FROM
    (
        SELECT Q.Q_NUM AS qNum, Q.Q_SUBJECT AS qSubject, Q.MAINTTEXT AS qMaintext, Q.VIEWS AS qViews, TO_CHAR(Q.ISUDATE, 'YYYY-MM-DD') AS qIsudate, S.USERID
        FROM QUESTION Q LEFT JOIN SIGNUP S
        ON Q.USER_NUM = S.USER_NUM
        WHERE Q.Q_SUBJECT LIKE '%||#{searchValue}||%'
    ) DATA
    WHERE RNUM BETWEEN #(start) AND #(end)
    ORDER BY RNUM DESC
```



12. 슬라이드 : 코딩설명

화면
발표 대본



MyBatis \${} 과 #{} 의 차이

WHERE \${searchKey} LIKE '%'||#{searchValue}||'%'

#{}
 파라미터가 String 형태로 들어와 자동적으로 '파라미터' 형태가 된다.

WHERE \${searchKey} LIKE '펫시터'



MyBatis \${} 과 #{} 의 차이

WHERE \${searchKey} LIKE '%'||#{searchValue}||'%'

WHERE SUBJECT LIKE '%'||#{searchValue}||'%'
 \${}

WHERE SUBJECT LIKE '%'||#{searchValue}||'%'
 해당 컬럼의 자료형에 맞추어 파라미터의 자료형의 변경된다.
 WHERE MAINTEXT LIKE '펫시터'#{searchValue} 사용한다.



13. 슬라이드 : 코딩설명

11

자유게시판 기능 (댓글 / 대댓글)

```
SELECT CO
FROM Q_CO
ON QC.Q_N
WHERE Q.Q
START WIT
CONNECT B
```

우리집 도마뱀이 밥을 먹었어요			
작성자	DOCASNAVAS	리뷰수	1
등록일	2019-12-16	조회수	906

2달 전 등록되었습니다. 마음에 들지 않으면 신고를 할 수 있습니다.

), QC.MAINTEXT))

_PARENT_NUM



알아있는 내용을 기반으로
~30%의 수익
~30%의 수익
~30%의 수익
~30%의 수익
~30%의 수익
~30%의 수익
~30%의 수익
~30%의 수익

화면

발표 대본

2

체크리스트 수행여부

JSP

```
<!-- 볼러를 때 이미 체크 했으면 checked 속성을 부여하고 disabled를 해야 함.. -->
<input id="box${status.index+1}" class="css-checkbox box${list.c_c_num}" type="checkbox"
onclick="check(${list.c_c_num})"
${list.excute == 'Y' ? "checked='checked'" : "" }${list.excute == 'Y' ? "disabled='disabled'" : "" }>
<label for="box${status.index+1}" class="css-label-red"></label>
```

jQuery

```
function check(c_c_num)
{
    var really = confirm("적용하시겠습니까? 한 번 체크한 항목은 취소할 수 없습니다.");
    var f = document.getElementById("check_form"+c_c_num);

    if(really == true)
    {
        var isChecked = $(".box"+c_c_num).is(":checked");
        if(isChecked == true)
        {
            $(".#excute"+c_c_num).val("Y");
        }
        else
        {
            $(".#excute"+c_c_num).val("N");
        }
        f.submit();
    }
    else
    {
        if($(".box"+c_c_num).is(":checked"))
        {
            $(".box"+c_c_num).prop("checked", false);
        }
        else
        {
            $(".box"+c_c_num).prop("checked", true);
        }
        return;
    }
}
```



14. 슬라이드 : 코딩설명

화면
발표 대본

2

체크리스트 수행여부

controller

```
@RequestMapping(value = "/checklistcheckok.action", method = {RequestMethod.GET, RequestMethod.POST})
public String checkok(Ccl_requestDTO dto, Model model, Pst_reserveDTO dto2)
{
    String view = null;

    ICcl_requestDAO dao = sqlSession.getMapper(ICcl_requestDAO.class);

    dto.setUser_num(sessionInfo.getUser_num());
    dto2.setUser_num(sessionInfo.getUser_num());
    dto2.setR_a_num(dto.getR_a_num());
    model.addAttribute("dto", dto2);

    dao.checkok(dto);

    view = "petsitterreservationnowdetail.action";
    return view;
}
```

Query(dao.xml)

```
<select id="p_sitting_checklist" resultType="com.team.petriots.ccl_request.dto.ccl_requestDTO">
    SELECT CC.C_C_NUM AS "C_C_NUM", RA.R_A_NUM AS "R_A_NUM", CC.EXCUTE AS "EXECUTE", CC.VIDLINK AS "VIDLINK"
    , CC.INGLINK AS "INGLINK", TO_CHAR(CC.TIME, 'HH24:MI') AS "TIME"
    , TO_CHAR(CR.CKSTDATE, 'HH24:MI') AS "STDATE", TO_CHAR(CR.CKENDATE, 'HH24:MI') AS "ENDDATE", CR.MAINTX AS "MAINTX"
    , CR.C_REQ_NUM AS "C_REQ_NUM"
    FROM CCL_REQUEST CR, CCL_APPROVE CA, CCL_CONFIRM CC, RSV_APPLY RA
    WHERE RA.R_A_NUM = CR.R_A_NUM
    AND CR.C_REQ_NUM = CA.C_REQ_NUM
    AND CA.C_A_NUM = CC.C_A_NUM
    AND RA.R_A_NUM = #{r_a_num}
</select>
```



15. 슬라이드 : 향후계획 & Q&A



향후계획

지금까지 구현한 미흡한 부분과 질문사항을 참고하여
프로젝트가 끝남에도 주기적으로 만나 수정 보완 해서
보다 발전된 웹 어플리케이션 작품을 완성할 예정입니다.



화면
발표 대본



기술문서

Location-Based WebApplication

AnswerController.java

- 설명
답변 테이블 컨트롤러로 답변폼에서 넘겨받은 번호와 페이지 번호를 수신하여 답글을 달게 한다
- 요구사항
IAnswerDAO.java, AnswerDTO, answer.xml
- 참고사항
페이지에서 질문번호를 넘겨받아 세팅하여 답변은 인서트한다

IAnswerDAO.java

- 설명
답변에 대한 추상 메소드(답글 리스트, 답글 작성, 답글 수정, 답글 삭제 등)를 구성한다
- 요구사항
페이지의 질문번호 qNum
- 참고사항
질문에대한 답글가져오기, 답글(작성,수정,삭제), 게시물읽기, 닉네임 번호로 찾기로 구성

AnswerDTO.java

- 설명
답변테이블과 답변댓글테이블의 getter / setter 구성
- 요구사항

bigDoor.jsp (대문페이지)

- 설명 : 메인페이지 접속전 보여줄 대문페이지
- 요구사항

js 파일: `<script src="BigDoorjs/jquery.min.js"></script>`
`<script src="BigDoorjs/jquery.scrolly.min.js"></script>`
`<script src="BigDoorjs/jquery.scrollex.min.js"></script>`
`<script src="BigDoorjs/skel.min.js"></script>`
`<script src="BigDoorjs/bigDoorutil.js"></script>`
`<script src="BigDoorjs/main.js"></script>`

CSS 파일: `main.css`

- 참고사항

`<div class="inner">` : 배너사진 들어갈 부분
`<div class="content">` : 배너 문구 들어갈 부분

adminChecklistModify.jsp (대문페이지)

- 설명 : 관리자 체크리스트 기본값 수정 페이지
- 요구사항

js 파일: `<script src="BigDoorjs/jquery.min.js"></script>`
`<script src="BigDoorjs/jquery.scrolly.min.js"></script>`
`<script src="BigDoorjs/jquery.scrollex.min.js"></script>`
`<script src="BigDoorjs/skel.min.js"></script>`
`<script src="BigDoorjs/bigDoorutil.js"></script>`
`<script src="BigDoorjs/main.js"></script>`

CSS 파일: `main.css`

- 참고사항

`<div class="panel-group">` : 체크리스트 관리 들어갈 부분
`<div class="panel-body">` : 기본값 항목, 체크리스트 기본값 들어갈 부분

petsitterApproveReservationDetail.jsp

- 설명 : 예약승낙페이지
- 요구사항

js 파일:

`<script src="vendor_mypage/jquery/jquery.min.js"></script>`
`<script src="vendor_mypage/bootstrap/js/bootstrap.bundle.min.js"></script>`

CSS 파일: `bootstrap.css`

- 참고사항

`<div class="container">` : 콘텐츠 내용 부분
`<div class="card-body">` : 예약 승낙, 내용, 요청자 정보 카드 구성 부분
`<div class="card-header">` : 위탁 번호 들어갈 부분

petsitterMain.jsp

- 설명 : 펫시터 모드 메인 페이지

- 요구사항

js 파일 :

jquery-ui.min.js, bootstrap.bundle.min.js, 1.7.1/jquery.min.js, query.min.js

CSS 파일 :

hyunwoo.css

bootstrap.css

- 참고사항

`div class="container"`

- 메인콘텐츠 들어갈 부분

`form id="user_form"`

- 완료 펫시팅 횟수가 들어갈 부분

`div class="card_info_section-3"`

- 프로필 수정 카드부분이 들어갈 부분

`div class="col-lg-12"`

- 현재 위탁중 정보가 들어갈 부분

`div class="card-media-body"`

- 위탁자 정보가 들어갈 부분

petsitterNowReservationDetail.jsp

- 설명 : 펫시터 모드 현재 예약 정보

- 요구사항

js 파일 :

jquery-ui.min.js

bootstrap.bundle.min.js

3.4.1/jquery.min.js

js/bootstrap.min.js

CSS 파일 :

hyunwoo.css

bootstrap.css

mainpage.css

- 참고사항

`div class="container"`

- 메인콘텐츠 들어갈 부분

`div class="card-header"`

- 예약 번호가 들어갈 부분

`div class="form-group"`

- 위탁 날짜, 체크인 시간, 체크 아웃 시간의 정보와 사용자가 요청한 서비스 (체크리스트) 내용이 들어갈 부분

`div class="card-media-body"`

- 회원 주소, 회원 아이디, 반려동물 정보가 들어갈 부분

petsitterNowSittingDetail.jsp

- 설명 : 펫시터 측면 현재 위탁 상세 정보 페이지
- 요구사항

js 파일 :

jquery.min.js

bootstrap.bundle.min.js

CSS 파일 :

hyunwoo.css

bootstrap.css

- 참고사항

`div class="container"`

- 메인콘텐츠 들어갈 부분

`div class="card-body"`

- 요청자 정보, 반려동물 사진이 들어갈 부분

`div class="form-group"`

- 위탁 날짜, 체크인/아웃 시간 위탁완료, 핫라인 메시지 보내기 등 버튼 기능이있는 부분

petsitterPastChecklist.jsp

- 설명 : 과거 체크리스트
- 요구사항

js 파일 :

jquery.min.js

bootstrap.bundle.min.js

1.8.18/jquery-ui.min.js

CSS 파일 :

hyunwoo.css

mainpage.css

- 참고사항

`div class="container"`

- 메인콘텐츠 들어갈 부분

`div class="card-header"`

- 나를 찜한사람 목록이 들어갈 부분

`div class="card"`

- 찜한 사람이 없을 때 출력할 카드 부분

petsitterBlacklist.jsp

- 설명 : 예약 승낙 페이지
- 요구사항

js 파일 :

jquery.min.js
bootstrap.bundle.min.js
3.4.1/jquery.min.js

CSS 파일 :

hyunwoo.css
bootstrap.css

- 참고사항

`div class="container"`

- 메인콘텐츠 들어갈 부분

`div class="details"`

- 블랙리스트 들어갈 부분

`div class="card-header"`

- 블랙리스트 목록 들어갈 부분

petsitterLikeList.jsp

- 설명 : 펫시터가 나를 즐겨찾기 한 사람목록 페이지
- 요구사항

js 파일 :

jquery.min.js
bootstrap.bundle.min.js
1.8.18/jquery-ui.min.js

CSS 파일 :

hyunwoo.css
mainpage.css

- 참고사항

`div class="container"`

- 메인콘텐츠 들어갈 부분

`div class="card-header"`

- 나를 찜한사람 목록이 들어갈 부분

`div class="card"`

- 찜한 사람이 없을 때 출력할 카드 부분

AnswerController.java

- 설명
답변 테이블 컨트롤러로 답변폼에서 넘겨받은 번호와 페이지 번호를 수신하여 답글을 달게 한다
- 요구사항
IAnswerDAO.java, AnswerDTO, answer.xml
- 참고사항
페이지에서 질문번호를 넘겨받아 세팅하여 답변은 인서트한다

IAnswerDAO.java

- 설명
답변에 대한 메소드(답글 리스트, 답글 작성, 답글 수정, 답글 삭제 등)를 구성한다
- 요구사항
페이지의 질문번호 qNum
- 참고사항
질문에대한 답글가져오기, 답글(작성,수정,삭제), 게시물읽기, 닉네임 번호로 찾기로 구성

AnswerDTO.java

- 설명
답변 테이블과 답변 댓글 테이블의 getter / setter 구성
- 요구사항
- 참고사항

// ANSWER 테이블

```
private int qNum ,aNum, aUserNum;  
private String aDelDate, aIsudate, aMaintext, userId;
```

// A_COMMENT 테이블

```
private int aCNum, aCUserNum, aCParentNum;  
private String aCMaintext, aCDelDate, aCIsudate;
```

answer.xml

- 설명
답글 insert, 답글 수정, 답글 삭제, 게시물읽기, 닉네임 번호로 찾기 쿼리문 구성
 - 요구사항
A_NUM(답변번호), Q_NUM(질문번호), USER_NUM(유저번호), MAINTEXT(내용)을 입력받아야 한다
 - 참고사항
Q_NUM(리스트 출력), MAINTEXT(인서트, 수정), A_NUM(삽입, 삭제, 게시물읽기)
-

ccl_d_rController.java

- 설명
체크리스트 기본값 출력 및 수정 삭제를 수행하는 컨트롤러
 - 요구사항
ICcl_d_rDAO.java, Ccl_d_rDTO.java, Ccl_d_rDAO.xml
 - 참고사항
c_d_r_num(기본값 항목 등록번호)를 입력받아 기본값을 삭제한다
-

ICcl_d_rDAO.java

- 설명
체크리스트 기본값에 대한 메소드(체크리스트 기본값 출력, 수정, 삭제)로 구성되어있다.
 - 요구사항
c_d_r_num(기본값 항목 등록번호)를 입력받아야 삭제가 가능하다
 - 참고사항
체크리스트 기본값 출력 및 수정 삭제 수행을 위한 메소드 구성
-

Ccl_d_rDTO

- 설명

체크리스트 기본값 테이블 변수에 대한 getter / setter 구성

- 요구사항

- 참고사항

```
private int c_d_r_num, c_d_num;  
private String maintext;
```

Ccl_d_rDAO.xml

- 설명

체크리스트 내용 출력과, 삭제, 수정에 대한 쿼리문 구성

- 요구사항

C_D_R_NUM(체크리스트 기본값 번호)가 있어야 삭제, 수정이 가능하다

- 참고사항

C_D_R_NUM(체크리스트 기본값 번호)로 수정과 삭제가 이루어짐

Ccl_requestController.java

- 설명

요청중 체크리스트 항목에 대한 값을 세션으로 로그인 중인 유저의 값을 입력받아 정보를 출력하는 컨트롤

- 요구사항

세션에 해당 유저에 대한 User_num을 가져와 요청 체크리스트를 사용 할 수 있다.

- 참고사항

세션에 해당 유저의 번호를 가져올 수 있어야 한다.

ICcl_requestDAO.java

- 설명
요청중 체크리스트 테이블에 대한 현재 위탁중인 체크리스트를 출력한다
- 요구사항
Ccl_requestDTO 가 구성되어 있어야 한다
- 참고사항
요청중 체크리스트에서의 현재 위탁중인 체크리스트를 출력한다

Ccl_requestDTO.java

- 설명
요청중 체크리스트에 대한 변수 getter / setter 구성
- 요구사항
- 참고사항
String r_a_num, c_c_num, c_req_num;
String excute, vidlink, imglink, time;
String sttdate, enddate, maintext;
String user_num;

Ccl_requestDAO.xml

- 설명
확정 체크리스트 항목 테이블의 번호를 통해 리스트 실행 여부를 업데이트 한다
- 요구사항
c_c_num(확정 체크리스트 항목 번호)를 입력받아야 한다
- 참고사항
요청 체크리스트 항목의 예약 작성 번호를 통해 리스트를 출력한다

cutlineController.java

- 설명
인성시험 및 지식시험에 대한 커트라인 리스트출력과, 커트라인수정, 삭제에 대한 컨트롤러
- 요구사항
cutline_num0이 삭제 메소드에 필요하다
- 참고사항
cutline_num으로 삭제 메소드 실행

ICutlineDAO.java

- 설명
커트라인 테이블에 대한 메소드 구성
- 요구사항
cutline_num 삭제메소드에 필요
- 참고사항
커트라인 테이블의 리스트 출력 수정 삭제에 대한 메소드를 구성한다

CutlineDTO

- 설명
커트라인에 대한 변수 getter / setter 구성
- 요구사항
- 참고사항
private int cutline_num, admin_num, p_cl, i_cl;
private String isudate;

CutlineDAO.xml

- 설명
체크리스트 출력, 체크리스트 삭제, 체크리스트 수정 쿼리문으로 구성되어 있다.
 - 요구사항
cutlien_num를 넘겨받아 메소드를 처리하므로 필요
 - 참고사항
cutline_num으로 리스트의 정렬을 하고 수정과 삭제에 조건절에 필요하다
-

distanceController.java

- 설명
거리에 관한 값을 출력하기 위한 컨트롤러
 - 요구사항
유저의 위치와 펫시터의 위치를 입력받아 거리값을 구한다
 - 참고사항
-

EntranceController.java

- 설명
페이지 이동을 위한 클래스로 대문페이지 이동, 메인페이지 이동에 대한 페이지 요청처리로 구성되어 있다.
 - 요구사항
 - 참고사항
["/bigdoor.action"](#)=대문페이지액션처리, ["/mainpage.action"](#)=메인페이지액션처리
-

I_q_bController.java

- 설명
지식문제 관련 컨트롤러로 지식문제 리스트 출력, 삽입, 수정, 삭제처리를 컨트롤 한다
- 요구사항
지식문제 번호인 num을 받아야 한다
- 참고사항
지식검사문제 삭제시 num(지식검사문제번호)를 입력받아 삭제 메소드를 수행한다

II_q_bDAO.java

- 설명
지식검사문제 테이블에 대한 인터페이스로 리스트출력, 삽입, 수정, 삭제에 대한 인터페이스
- 요구사항
지식검사문제 번호를 받아야 한다
- 참고사항
삭제 메소드의 경우 지식검사문제 번호로 삭제 메소드를 수행한다

I_q_bDTO.java

- 설명
지식검사 시험 테이블의 getter / setter 구성
- 요구사항
- 참고사항

```
private int num, admin_num, correct;
private String maintext, chioce1, chioce2, chioce3, chioce4, chioce5;
```

I_q_bDAO.xml

- 설명
지식검사문제 리스트, 삽입, 수정, 삭제 쿼리문을 구성하여 쿼리를 실행한다
- 요구사항
지식검사문제 번호(num), 내용(maintext), 정답(correct) 등 수정과 삭제를 위해 필요
- 참고사항
지식검사문제 번호(num)은 수정과 삭제 쿼리문에 사용된다

ImageDAO.java

- 설명
이미지 수정에 대한 메소드 구성
- 요구사항
사진을 업데이트 시키기 위한 쿼리문 구성
- 참고사항
USER_NUM(유저번호), P_R_NUM(반려동물 등록번호), P_S_NUM(펫시터 전환번호) 필요

ImageServlet.java

- 설명
세션에 이미지를 담아 유저의 프로필 이미지를 유지
- 요구사항
해당 유저의 user_num 와 p_r_num이 필요하다
- 참고사항
해당 이미지의 최대 파일 사이즈는 10*1024*1024이다.

Mtom_qController.java

- 설명

1대1문의 컨트롤러로 세션의 리스트출력, 수정 액션처리를 한다

- 요구사항

해당 1대1 문의 메소드를 수행하기 위해 세션에 해당 유저의 User_num이 필요하다

- 참고사항

`dto.setUser_num(SessionInfo.getUser_num());`으로 해당 유저의 User_num을 set한다

IMtom_qDAO.java

- 설명

1대1문의 인터페이스로 리스트출력, 수정 등의 메소드가 존재한다

- 요구사항

Mtom_qDTO

- 참고사항

`public int sendquestion(Mtom_qDTO dto);` // 문의하기 메소드

Mtom_qDTO.java

- 설명

1대1문의 테이블의 dto로 getter / setter 구성

- 요구사항

1대1문의 테이블과 1대1문의 답글 테이블의 getter / setter 구성

- 참고사항

문의번호, 유저번호, 문의일자, 삭제일자, 문의분류번호, 문의분류와 답글테이블로 구성

Mtom_qDAO.xml

- 설명
1대1문의 테이블 리스트, 수정 쿼리문 구성 dao.xml
 - 요구사항
1대1 문의 테이블과 1대1 답글 테이블을 조인하여 리스트 출력
 - 참고사항
MTOM_Q, MTOM_A 테이블 조인
-

NoticeController.java

- 설명
공지사항 리스트출력 및 게시판 구성 컨트롤러
 - 요구사항
게시물 이용시 해당 게시물의 페이지 넘버로 게시물의 번호 확인 하여 게시물 요청을 처리
 - 참고사항
페이징 처리를 통하여 게시물의 리스트를 출력한다
-

INoticeDAO.java

- 설명
공지사항 게시판 구성을 위한 notice 테이블 getter / setter 구성
 - 요구사항
@Param으로 페이징 처리
 - 참고사항
ntcNum(공지글 번호)로 삭제 수정 메소드 수행
-

NoticeDTO.java

- 설명
공지사항 게시판 구성을 위한 변수 getter / setter 구성
 - 요구사항
Notice 테이블과 게시글 번호 속성인 rNum(게시글 번호 속성)을 구성
 - 참고사항
int rNum은 게시글의 번호 속성을 나타내는 변수이다.
-

Notice.xml

- 설명
공지사항 게시판의 게시판 페이지 처리와 수정, 삭제등을 위한 쿼리문 구성 xml
 - 요구사항
게시판 페이지 처리를 위해 NTC_NUM을 내림차순으로 처리
 - 참고사항
ASC 오름차순, DESC 내림차순
-

P_q_bController.java

- 설명
인성문제 관련 컨트롤러로 문제 리스트, 삽입, 수정, 삭제 기능을 갖춤
 - 요구사항
p_t_q_num으로 삭제 메소드 수행
 - 참고사항
-

IP_q_bDAO.java

- 설명
인성문제 인터페이스로 리스트 출력, 수정, 삭제 메소드 등을 선언
- 요구사항
삭제 메소드 수행시 int p_t_q_num을 받아 수행
- 참고사항

P_q_bDTO.java

- 설명
인성검사 테이블로 변수 getter / setter 구성
- 요구사항
- 참고사항
p_t_q_num, correct, maintext, chioce1 2 3 4 5, admin_num 으로 구성

P_q_bDAO.xml

- 설명
인성문제 리스트, 삽입, 수정 ,삭제를 위한 쿼리문 구성
- 요구사항
인성문제 테이블의 p_t_q_num을 가져와야함
- 참고사항

Pet_registerController.java

- 설명
반려동물 등록 테이블 컨트롤러로 등록시 해당 로그인된 사용자의 정보에 등록됨
 - 요구사항
세션의 로그인 정보
 - 참고사항
세션으로 로그인된 사용자가 반려동물 추가시 해당 사용자에게 등록됨
-

IPet_registerDAO.java

- 설명
반려동물 등록 인터페이스로 수정 삭제 삽입 등과 같은 메소드 선언
 - 요구사항
Pet_registerDTO, user_num, p_r_num
 - 참고사항
-

Pet_registerDTO.java

- 설명
반려동물 등록 dto로 해당 테이블 getter / setter 구성
 - 요구사항
 - 참고사항
p_r_num, name, isudate, user_num, p_t_num 등 변수 구성
-

Pet_registerDAO.xml

- 설명
반려동물 등록 테이블의 리스트출력, 수정, 삭제 메소드 구성
- 요구사항
p_r_num, user_num 값 필요
- 참고사항

Pst_profileController.java

- 설명
프로필 등록 테이블 컨트롤러로 예약사항을 조회하여 체크인 여부등 업무 수행
- 요구사항
SignupDTO, Pst_profileDTO
- 참고사항
나를 찜한 사람 리스트와 블랙리스트 삭제등의 컨트롤러

IPst_profileDAO.java

- 설명
프로필 등록 테이블 인터페이스로 펫시터의 예약 요청에 관련된 메소드 구성
- 요구사항
Pst_profileDTO, p_s_num, user_num
- 참고사항
펫시터 요청된 예약, 찜, 블랙리스트, 위탁수, 업적등의 메소드 선언

Pst_profileDTO.java

- 설명
프로필 등록 테이블 변수 getter / setter 구성
- 요구사항
프로필 등록 테이블 모든 변수 구성
- 참고사항
isupdate = 블랙리스트 등록 날짜 , addrLat, addrLong, distnace = 위경도 좌표

Pst_profileDAO.xml

- 설명
프로필 등록 테이블 dao.xml로 쿼리문 구성 xml
- 요구사항
Pst_profileDTO
- 참고사항
블랙리스트 조회 쿼리문,블랙리스트 삭제 쿼리문, 펫시터 조회 리스트 등

Pst_shiftController.java

- 설명
펫시터 전환 테이블
- 요구사항
CutlineDTO
- 참고사항

IPst_shiftDAO.java

- 설명

펫시터 전환 테이블 인터페이스 구성으로 전환을 위한 메소드 구성

- 요구사항

user_num, question, c_p_t_num, s_p_t_num, m_p_t_num, cutline_num, p_p_t_num

- 참고사항

Pst_shiftDAO.xml

- 설명

펫시터 전환을 위한 쿼리문 구성 dao.xml

- 요구사항

Marking_qDTO, CutlineDTO, Solve_qDTOM Couduct_qDTOM P_q_bDTO, Pass_pDTO

- 참고사항

Couduct_pDTO

- 설명

인성검사 시행테이블로 인성검사 시행에 필요한 변수 getter / setter 구성

- 요구사항

c_p_t_num, user_num, isudate, q1~20

- 참고사항

q1~20 = q1부터 q20 번까지를 나타냄

CutlineDTO.java

- 설명
커트라인 DTO로 인성검사 커트라인과 지식검사 커트라인 구성을 위한 getter / setter 구성
- 요구사항
cutline_num, admin_num, p_cl, l_cl, isudate
- 참고사항

Marking_pDTO

- 설명
인성검사 채점 테이블로 채점에 필요한 변수 getter / setter 구성
- 요구사항
m_p_t_num, s_p_t_num, t1~20
- 참고사항
t1부터 t20 까지는 인성문제 채점

P_q_bDTO.java

- 설명
인성검사 문제 구성 테이블로 해당 테이블 getter / setter 구성
- 요구사항
p_t_q_num, admin_num, mainText, correct, chioce1~ 5
- 참고사항
chioce1~5는 보기 1~5를 나타낸다.

Pass_pDTO.java

- 설명
인성검사 합격 테이블로 해당 테이블의 getter / setter 구성
- 요구사항
p_p_t_num, m_p_t_num, isudate, cutline, getP_p_t_num()
- 참고사항

Pass_tDTO.java

- 설명
최종합격 테이블 dto로 해당 테이블 getter / setter 구성
- 요구사항
p_s_num, p_p_t_num, p_i_t_num, isudate, getP_s_num()
- 참고사항

Pst_shiftDTO.java

- 설명
펫시터 전환 테이블 dto로 해당 테이블 getter / setter 구성
- 요구사항
p_s_num, user_num
- 참고사항

Solve_pDTO.java

- 설명
인성검사 풀이번호 테이블 dto로 getter / setter 구성
- 요구사항
s_p_t_num, c_p_t_num, isudate, a1~20
- 참고사항
a1~20까지는 인성검사 답안 20번 까지를 말한다

QuestionController.java

- 설명
질문 테이블 컨트롤러로 질문게시판 구성을 위한 메소드 컨트롤
- 요구사항
공지사항 게시판과 마찬가지로 페이지 처리 필요
- 참고사항

IQuestionDAO.java

- 설명
질문 테이블 인터페이스로 Q&A 게시판 구성을 위한 메소드 선언
- 요구사항
데이터 갯스 파악을 위한 @param("searchKey")등 메소드 선언
- 참고사항

QuestionDTO.java

- 설명
질문 테이블 dto로 해당 테이블 getter / setter 구성
- 요구사항
qNum, qViews, qUserNum, qDeldDate, qIsudate, qMaintext, userId
qCParentNum, qCNum, qCUserNum, qCMaintext, qCDelDate, qCIsudate 등 구성
- 참고사항

question.xml

- 설명
게시판 구성을 위해 쿼리문 구성하는 xml
- 요구사항
QuestionDTO,
- 참고사항
질문에 달린 댓글, 답글의 DELDATE를 업데이트하여 보이지 않게 한다

Reg_my_cclDTO.java

- 설명
체크리스트번호 등 변수 getter / setter 구성
- 요구사항
r_m_c_num, p_r_num, sttime, endtime, maintxt
- 참고사항

register_AdminController.java

- 설명
관리자 테이블로 관리자 로그인액션과 로그아웃 액션처리
- 요구사항
- 참고사항
로그아웃 시에 session.removeAttribute("admin_num")

IRegister_adminDAO.java

- 설명
관리자 테이블 인터페이스로 로그인 관련 메소드 선언
- 요구사항
Register_adminDTO
- 참고사항
Register_adminDTO 와 Register_adminSessionInfo로 구성되어 있음

Register_adminDTO.java

- 설명
관리자 테이블 dto로 해당 테이블 getter / setter 구성
- 요구사항
admin_num, id, pw
- 참고사항

Register_adminSessionInfo.java

- 설명

관리자 테이블에서의 로그인 세션 구현에 대한 dto로 admin_num은 스택틱 변수로 구성

- 요구사항

private static int admin_num

- 참고사항

Register_adminDAO.xml

- 설명

관리자 테이블의 로그인을 통하여 세션기능을 수행하기 위한 쿼리문으로 구성

- 요구사항

select result type = java.lang.Integer 원시타입으로 구성

- 참고사항

ReportController.java

- 설명

신고 테이블의 컨트롤러로 신고 리스트, 신고 삽입(신고처리)로 구성되어 있다

- 요구사항

- 참고사항

declarationupdate.action은 신고 처리에 의한 액션 네임명이다

IReportDAO.java

- 설명
신고테이블 인터페이스로 신고리스트, 신고처리 메소드로 구성 되어있다
- 요구사항
ReportDTO
- 참고사항

ReportDTO.java

- 설명
신고 테이블의 dto로 해당 테이블의 getter / setter 구성으로 되어있다
- 요구사항
rpt_num, reporter, reported, rpt_type, sct_num, admin_num, sct_type_num, sctday
- 참고사항

ReportDAO.xml

- 설명
신고 테이블의 리스트 출력과 신고처리를 위한 쿼리문 구성
- 요구사항
ReportDTO
- 참고사항
인서트 수행시 신고에 대한 값을 삽입하여 신고처리를 진행

SignupController.java

- 설명
회원 테이블 컨트롤러로 회원 수정, 회원가입 폼을 통해 가입을 진행하여 처리하는 컨트롤러
- 요구사항
해당 유저가 세션에 로그인한 값이 저장되어 있어야 한다
- 참고사항
회원 가입 및 정보 삭제 수정과 같은 기능도 포함되어 있다.

ISignupDAO.java

- 설명
회원테이블 인터페이스로 회원리스트출력, 수정, 로그인, 관리자영역 으로 구성되어 있다.
- 요구사항
SignupDTO, user_num
- 참고사항
회원 테이블의 관리자가 수정하는 영역은 따로 존재한다.

SignupDTO.java

- 설명
회원테이블 dto로 해당 테이블의 getter / setter 구성
- 요구사항
user_num, pwd, pwd2, userId, uwerName, ssn, zipcode, address, addrdetails 등
- 참고사항
addrLat, addrLong 은 위경도 좌표로 회원의 위치로 거리를 구하기 위해 존재한다.

MemberServlet.java

- 설명
회원 로그인에 대한 처리를 위한 구성으로 로그인관련 메소드가 존재한다
- 요구사항
login.jsp로부터 입력받은 값을 메소드를 통해 검사하여 액션처리
- 참고사항
로그인 성공시 세션에 값을 저장함

MemberDTO.java

- 설명
회원예 대한 정보를 구성한 dto로 getter / setter를 구성
- 요구사항
위경도에 대한 변수는 자료형 double로 구성한다
- 참고사항
addrLat, addrLong = 주소 위경도

MyUtil.java

- 설명
게시판에 대한 페이징 처리를 위한 클래스
- 요구사항
- 참고사항

SessionInfo.java

- 설명

회원 세션 유지를 위해 구성된 dto로 static String 과 double 자료형으로 구성

- 요구사항

```
private static String user_num, p_s_num, address, pwd;
```

```
private static double addrLat, addrLong;
```

- 참고사항

DPCPConn.java

- 설명

데이터 베이스에 연결하는 구문이 작성된 클래스

- 요구사항

- 참고사항

SingupDAO.xml

- 설명

회원과 관련된 수정 로그인에 대한 처리를 위한 쿼리 구성

- 요구사항

SignupDTO

- 참고사항

IPst_reservController.java

- 설명
승인하거나 현재 위탁건에 대한 정보들을 컨트롤하는 클래스
- 요구사항
Pst_reserveDTO, Ccl_requestDTO
- 참고사항

IPst_reserveDAO.java

- 설명
예약신청과 현재와 지난 위탁건에 대한 메소드 선언
- 요구사항
Pst_reserveDTO
- 참고사항

Pst_reserveDTO.java

- 설명
펫시터 측면에서 위탁 / 예약 중인 카드 정보를 저장하는 DTO
- 요구사항
user_num, nickName, p_r_num, profile_img, pet_img, r_a_num, address, pet_name 등
- 참고사항

Pst_reserveDAO.xml

- 설명
위탁에 대한 건과 예약에 관한 쿼리를 구성
- 요구사항
Pst_reserveDTO
- 참고사항

WithdrawalController.java

- 설명
회원탈퇴 테이블로 회원 삭제가 가능하다
- 요구사항
WithdrawalDTO
- 참고사항

IWithdrawalDAO.java

- 설명
회원탈퇴 테이블의 리스트와 삽입 메소드 선언
- 요구사항
WithdrawalDTO dto
- 참고사항

WithdrawalDTO.java

- 설명
회원 탈퇴 테이블로 관련된 getter / setter 구성
- 요구사항
wd_num, isudate, wd_r_num, details, user_num, pwd 로 구성
- 참고사항

WithdrawalDAO.xml

- 설명
회원 탈퇴 테이블의 쿼리문 구성으로 이루어져
 - 요구사항
WithdrawalDTO
 - 참고사항
-
-

noticeList.jsp (공지사항 리스트 출력)

- 설명 : 공지사항 리스트 출력 페이지

- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](http://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
 CSS 파일 : [notice_style.css](#)
[notice_created.css](#)
[mainpage.css](#)

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbsList_header"></code>	: 게시물의 상단
<code><div id="bbsList_list"></code>	: 본문
<code><div id="footer"></code>	: 게시물 하단

noticeCreated.jsp (공지사항 글 등록)

- 설명 : 공지사항 글 등록 페이지

- 요구사항

js 파일 : <http://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.js>
<http://netdna.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.js>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.js>
[util.js](#)
 CSS 파일 : <http://netdna.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.css>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.css>
[notice_style.css](#)
[notice_created.css](#)
[mainpage.css](#)

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbs_title"></code>	: 게시물 상단
<code><div id="bbsCreated_footer"></code>	: 게시물 하단

noticeArticle.jsp (공지사항 각 게시물 출력)

- 설명 : 공지사항 각 게시물 출력 페이지
- 요구사항

CSS 파일 : *notice_style.css*
notice_article.css
mainpage.css

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbs_title"></code>	: 게시물 상단
<code><div id="bbsArticle"></code>	: 게시물 본문 틀
<code><div id="bbsArticle_header"></code>	: 게시물 본문 상단
<code><div class="bbsArticle_bottomLine"></code>	: 게시물 본문 하단
<code><div id="bbsArticle_content"></code>	: 게시물 본문
<code><div class="bbsArticle_noLine"></code>	: 게시물 본문 선 제거
<code><div id="bbsArticle_footer"></code>	: 게시물 하단

noticeUpdated.jsp (공지사항 게시물 업데이트)

- 설명 : 공지사항 리스트 출력 페이지
- 요구사항

js 파일 : <http://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.js>
<http://netdna.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.js>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.js>
util.js

CSS 파일 : <http://netdna.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.css>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.css>
notice_style.css
notice_created.css
mainpage.css

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbs_title"></code>	: 게시물 상단
<code><div id="bbsCreated_footer"></code>	: 게시물 하단

questionList.jsp (Q/A 리스트 출력)

- 설명 : Q/A 리스트 출력 페이지

- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](http://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
 CSS 파일 : [notice_style.css](#)
[notice_created.css](#)
[mainpage.css](#)

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbsList_header"></code>	: 게시물의 상단
<code><div id="bbsList_list"></code>	: 본문
<code><div id="footer"></code>	: 게시물 하단

questionCreated.jsp (Q/A 글 등록)

- 설명 : Q/A 글 등록 페이지

- 요구사항

js 파일 : <http://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.js>
<http://netdna.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.js>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.js>
[util.js](#)
 CSS 파일 : <http://netdna.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.css>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.css>
[notice_style.css](#)
[notice_created.css](#)
[mainpage.css](#)

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbs_title"></code>	: 게시물 상단
<code><div id="bbsCreated_footer"></code>	: 게시물 하단

questionArticle.jsp (Q/A 각 게시물 출력)

- 설명 : Q/A 각 게시물 출력 페이지
- 요구사항

CSS 파일 : *notice_style.css*
notice_article.css
mainpage.css

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbs_title"></code>	: 게시물 상단
<code><div id="bbsArticle"></code>	: 게시물 본문 틀
<code><div id="bbsArticle_header"></code>	: 게시물 본문 상단
<code><div class="bbsArticle_bottomLine"></code>	: 게시물 본문 하단
<code><div id="bbsArticle_content"></code>	: 게시물 본문
<code><div class="bbsArticle_noLine"></code>	: 게시물 본문 선 제거
<code><div id="bbsArticle_footer"></code>	: 게시물 하단

questionUpdated.jsp (Q/A 게시물 업데이트)

- 설명 : Q/A 리스트 출력 페이지
- 요구사항

js 파일 : <http://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.js>
<http://netdna.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.js>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.js>
util.js

CSS 파일 : <http://netdna.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.css>
<http://cdnjs.cloudflare.com/ajax/libs/summernote/0.8.8/summernote.css>
notice_style.css
notice_created.css
mainpage.css

- 참고사항

<code><div id="bbs"></code>	: 게시물의 전체적인 틀
<code><div id="bbs_title"></code>	: 게시물 상단
<code><div id="bbsCreated_footer"></code>	: 게시물 하단

Mypage_fail_humanity.jsp (인성검사 탈락)

- 설명 : 인성검사 탈락 페이지

- 요구사항

js 파일 : *jquery.min.js*
bootstrap.bundle.min.js
https://fonts.googleapis.com/css?family=Raleway:300,700
 CSS 파일 : *bootstrap.css*
mainpage.css

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="col-lg-12"></code>	: 게시물 비율
<code><div class="card-media-body"></code>	: 게시물 본문

mypageApproveReservationDetail.jsp (예약 승낙)

- 설명 : 예약 승낙 페이지

- 요구사항

js 파일 : *jquery.min.js*
bootstrap.bundle.min.js
 CSS 파일 : *cards_hyunwoo.css*
bootstrap.css
https://fonts.googleapis.com/css?family=Raleway:300,700

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 단위(수)
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="card-body"></code>	: 프로필 카드 틀
<code><div class="card-media"></code>	: 프로필 카드 사진 틀
<code><div class="card-media-body"></code>	: 프로필 본문 틀

mypageFAQ.jsp (FAQ)

- 설명 : FAQ 메인 페이지

- 요구사항

js 파일 : *//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js*
//code.jquery.com/ui/1.8.18/jquery-ui.min.js
jquery.min.js
bootstrap.bundle.min.js
 CSS 파일 : *jquery-ui.css*
bootstrap.css

- 참고사항

<code><div class="faq_container"></code>	: FAQ 상단 메뉴 틀
<code><div class="faq"></code>	: FAQ 내용 틀

mypageLikeList.jsp (찜 목록)

- 설명 : 찜 목록 페이지

- 요구사항

js 파일 : <https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js>
[bootstrap.bundle.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)
[jquery-ui.css](#)
[mainpage.css](#)
[PetsitterQuestList.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<div class="container">	: 전체적인 틀
<div class="row">	: 게시물 배열
<div class="col-lg-12">	: 게시물 화면 비율
<div class="card-media-body">	: 게시물 본문

mypageMain.jsp (마이페이지)

- 설명 : 마이 페이지 출력

- 요구사항

js 파일 : <https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js>
[bootstrap.bundle.min.js](#)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](#)
[jquery.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)
[//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css](#)
[mainpage.css](#)

- 참고사항

<div class="container">	: 전체적인 틀
<div class="row">	: 게시물 배열
<div class="col-lg-12">	: 게시물 화면 비율
<div class="card-media-body">	: 게시물 본문

mypageNowSittingDetail.jsp (유저의 현재 위탁 정보)

- 설명 : 유저의 현재 위탁 상세 정보 출력 페이지
- 요구사항

js 파일 : *bootstrap.bundle.min.js*
 CSS 파일 : *cards_hyunwoo.css*
bootstrap.css
mainpage.css
https://fonts.googleapis.com/css?family=Raleway:300,700

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="card-media-body"></code>	: 게시물 본문
<code><div class="form-group"></code>	: 보조 내용 틀

mypagePastsittingDetail.jsp (과거 위탁 정보)

- 설명 : 과거 위탁 정보 출력 페이지
- 요구사항

js 파일 : *bootstrap.bundle.min.js*
jquery.min.js
 CSS 파일 : *cards_hyunwoo.css*
bootstrap.css
mainpage.css

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="card-media-body"></code>	: 게시물 본문
<code><div class="form-group"></code>	: 보조 내용 틀

mypagePetAdd.jsp (반려동물 추가)

- 설명 : 반려동물 추가 페이지
- 요구사항

js 파일 : *bootstrap.bundle.min.js*
jquery.min.js
jquery-ui.js
 CSS 파일 : *https://fonts.googleapis.com/css?family=Raleway:300,700*
bootstrap.css
mainpage.css

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="card-media-body"></code>	: 게시물 본문
<code><div class="form-group"></code>	: 보조 내용 틀

mypagePetList.jsp (반려동물 목록)

- 설명 : 현재 나의 반려동물 목록 출력 페이지
- 요구사항

js 파일 : [//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[jquery.min.js](#)
[bootstrap.bundle.min.js](#)

CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)
[mainpage.css](#)
[//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css](https://code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<div class="container">	: 전체적인 틀
<div class="row">	: 게시물 배열
<div class="col-lg-5">	: 게시물 화면 비율

mypagePetModify.jsp (반려동물 정보 수정)

- 설명 : 반려동물 정보 수정 페이지
- 요구사항

js 파일 : [jquery.min.js](#)
<https://code.jquery.com/jquery-1.12.4.js>
<https://code.jquery.com/ui/1.12.1/jquery-ui.js>
[bootstrap.bundle.min.js](#)

CSS 파일 : [//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css](https://code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css)
[bootstrap.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<div class="container">	: 전체적인 틀
<div class="row">	: 게시물 배열
<div class="card-header">	: 게시물 상단
<div class="card-media-body">	: 게시물 본문
<div class="form-group">	: 보조 내용 틀

mypageQForm.jsp (문의하기)

- 설명 : 문의하기 입력 틀 페이지
- 요구사항

js 파일 : *jquery.min.js*
bootstrap.bundle.min.js
 CSS 파일 : *mainpage.css*
bootstrap.css

- 참고사항

`<div class="container">` : 전체적인 틀

mypageRequestReservationDetail.jsp (유저 현재 예약 정보)

- 설명 : 유저 현재 예약 정보 출력 페이지
- 요구사항

js 파일 : *jquery.min.js*
bootstrap.bundle.min.js
bootstrap.min.js
 CSS 파일 : *cards_hyunwoo.css*
bootstrap.css
mainpage.css

- 참고사항

`<div class="container">` : 전체적인 틀
`<div class="row">` : 게시물 배열
`<div class="col-lg-12">` : 게시물 화면 비율
`<div class="card-media-body">` : 게시물 본문
`<div class="form-group">` : 보조 내용 틀

mypageReservationApprove.jsp (승인된 예약 내역)

- 설명 : 승인된 예약 내역 조회 정보 출력 페이지
- 요구사항

js 파일 : *jquery.min.js*
bootstrap.bundle.min.js
//code.jquery.com/ui/1.8.18/jquery-ui.min.js
 CSS 파일 : *//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css*
bootstrap.css
mainpage.css

- 참고사항

`<div class="container">` : 전체적인 틀
`<div class="row">` : 게시물 배열
`<div class="col-lg-12">` : 게시물 화면 비율
`<div class="card-media-body">` : 게시물 본문
`<div class="form-group">` : 보조 내용 틀

mypageReservationList.jsp (위탁 내역 조회)

- 설명 : 위탁 내역 조회 페이지

- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[jquery.min.js](#)
[bootstrap.bundle.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[mainpage.css](#)
[bootstrap.css](#)
[//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css](https://code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="card-media-body"></code>	: 게시물 본문
<code><div class="form-group"></code>	: 보조 내용 틀

mypageReservationNow.jsp (위탁 확인)

- 설명 : 현재 자신이 위탁한 반려동물 확인 페이지

- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[jquery.min.js](#)
[bootstrap.bundle.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[mainpage.css](#)
[bootstrap.css](#)
[//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css](https://code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="card-media-body"></code>	: 게시물 본문
<code><div class="form-group"></code>	: 보조 내용 틀
<code><div class="container_card"></code>	: 위탁 내용 표시 틀

mypageReservationRequest.jsp (예약 확인)

- 설명 : 찜 목록 페이지

- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[jquery.min.js](#)
[bootstrap.bundle.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)
[mainpage.css](#)
[//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css](https://code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css)

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="card-media-body"></code>	: 게시물 본문
<code><div class="form-group"></code>	: 보조 내용 틀
<code><div class="container_card"></code>	: 위탁 내용 표시 틀

myPageRuleForPetsitter.jsp (펫시터 전환 약관 동의)

- 설명 : 펫시터 전환 약관 동의 폼 출력 페이지

- 요구사항

js 파일 : [bootstrap.bundle.min.js](#)
<https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js>
 CSS 파일 : [bootstrap.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>
[mainpage.css](#)

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="col-lg-12"></code>	: 게시물 화면 비율
<code><div class="form-check"></code>	: 약관 동의 체크 틀

mypageSideMenu.jsp (마이페이지 사이드 메뉴)

- 설명 : 마이페이지의 여러 가지 기능을 모아둔 사이드 메뉴 페이지
- 요구사항
 - CSS 파일 : <https://maxcdn.bootstrapcdn.com/font-awesome/4.6.3/css/font-awesome.min.css>
- 참고사항
 - `<ul class="side-menu">` : 마이페이지 사이드 메뉴 위치 및 디자인

myPageTestPersonality.jsp (펫시터 인성 검사)

- 설명 : 펫시터 인성 검사 문제지 출력 페이지
- 요구사항
 - js 파일 : [jquery.min.js](#)
[bootstrap.bundle.min.js](#)
<https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js>
 - CSS 파일 : [bootstrap.css](#)
[mainpage.css](#)
- 참고사항
 - `<div class="container">` : 전체적인 틀
 - `<div class="row">` : 게시물 배열
 - `<div class="col-lg-12">` : 게시물 화면 비율
 - `<div class="submitter">` : 인성시험 문제 제출 버튼
 - `<div class="quizAnswers">` : 틀린 답 표시 틀

mypageUserModify.jsp (내 정보 수정)

- 설명 : 로그인 한 계정의 내 정보를 수정하는 페이지
- 요구사항
 - js 파일 : [jquery.min.js](#)
[bootstrap.bundle.min.js](#)
http://dmaps.daum.net/map_js_init/postcode.v2.js
<//code.jquery.com/ui/1.8.18/jquery-ui.min.js>
<//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js>
 - CSS 파일 : [bootstrap.css](#)
[mainpage.css](#)
[cards_hyunwoo.css](#)
<//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css>
- 참고사항
 - `<div class="container">` : 전체적인 틀
 - `<div class="row">` : 게시물 배열
 - `<div class="form-group">` : 보조 내용 틀

mypageWithdrawal.jsp (회원 탈퇴)

- 설명 : 회원 탈퇴 페이지

- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)
CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)
[//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css](https://code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css)
[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="form-group"></code>	: 보조 내용 틀
<code><div class="card-header"></code>	: 게시물 상단

petsitterPastSittingDetail.jsp

- 설명 : 펫시터 과거 위탁 상세 페이지

- 요구사항

js 파일 : [vendor_mypage/bootstrap/js/bootstrap.bundle.min.js](#)
[vendor_mypage/jquery/jquery.min.js](#)

CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)
[//code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css](https://code.jquery.com/ui/1.8.18/themes/base/jquery-ui.css)
[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="card-body"></code>	: 요청자 정보 내용이 담김
<code><div class="card-media"></code>	: 유저정보, 반려동물 정보가 담김
<code><div class="form-group"></code>	: 과거 위탁정보가 담감 (체크인/아웃) 등

petsitterProfileAdd.jsp

- 설명 : 펫시터_프로필 등록 페이지
- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)

CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="form-group"></code>	: 위탁 정보 설정 내용이 담김
<code><div class="card-header"></code>	: 게시물 상단

petsitterProfileModify.jsp

- 설명 : 펫시터_프로필 수정 테이블
- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)

CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열
<code><div class="form-group"></code>	: 위탁 정보를 수정하는 실 내용이 담김
<code><div class="card-header"></code>	: 게시물 상단

petsitterQuestList.jsp

- 설명 : 펫시터_업적 목록 페이지
- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="multipl-image-checkbox"></code>	: 업적 체크박스가 담김
<code><div class="form-group"></code>	: 위탁 정보 설정 내용이 담김
<code><div class="card-header"></code>	: 업적목록 출력하는 내용이 담김

petsitterRequestReservationDetail.jsp

- 설명 : 펫시터_ 현재 예약 정보 페이지
- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열, 예약 번호 출력
<code><div class="form-group"></code>	: 예약에 관한 상세 정보가 담김
<code><div class="card-header"></code>	: 게시물 상단

petsitterReservationList.jsp

- 설명 : 펫시터_지난 위탁 목록 출력 페이지
- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="card-media-body"></code>	: 위탁 유저와, 반려동물 정보가 담김
<code><div class="form-group"></code>	: 위탁 정보 설정 내용이 담김
<code><div class="card-header"></code>	: 업적목록 출력하는 내용이 담김

petsitterReservationNow.jsp

- 설명 : 펫시터_ 현재 위탁/예약 페이지
- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="row"></code>	: 게시물 배열, 예약 번호 출력
<code><div class="form-group"></code>	: 예약/위탁 관한 상세 정보(체크리스트)가 담김
<code><div class="card-header"></code>	: 게시물 상단

petsitterReservationRequest.jsp

- 설명 : 펫시터_예약 요청 확인 페이지
- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)
<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="card-media-body"></code>	: 위탁 유저와, 반려동물 정보가 담김
<code><div class="form-group"></code>	: 사용자가 요청한 체크리스트 내용이 담김
<code><div class="card-header"></code>	: 예약 요청 리스트 출력 내용이 담김

petsitterReserveCancelForm.jsp

- 설명 : 펫시터_예약 취소 페이지
- 요구사항

js 파일 : <https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js>

CSS 파일 : [ReviewForm.css](#)

- 참고사항

<code><div class="review-form"></code>	: 취소사유 전체 틀
--	-------------

petsitterReservationList.jsp

- 설명 : 펫시터_지난 위탁 목록 출력 페이지

- 요구사항

js 파일 : [//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js](https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js)
[//code.jquery.com/ui/1.8.18/jquery-ui.min.js](https://code.jquery.com/ui/1.8.18/jquery-ui.min.js)
[bootstrap.bundle.min.js](#)
[jquery.min.js](#)
 CSS 파일 : [cards_hyunwoo.css](#)
[bootstrap.css](#)

[mainpage.css](#)

<https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

<code><div class="container"></code>	: 전체적인 틀
<code><div class="card-media-body"></code>	: 위탁 유저와, 반려동물 정보가 담김
<code><div class="form-group"></code>	: 위탁 정보 설정 내용이 담김
<code><div class="card-header"></code>	: 업적목록 출력하는 내용이 담김

petsitterSideMenu.jsp

- 설명 : 펫시터_측면 사이드 메뉴

- 요구사항

js 파일 :

CSS 파일 : <https://fonts.googleapis.com/css?family=Raleway:300,700>

- 참고사항

`ul class="side-menu"` : 사이드 메뉴 전체적인 틀, 내용

파일구조도

Location-Based WebApplication

Depth1	Depth2	Depth3	비고
CSS	badges		펫시터의 벤티 부여에 사용될 이미지
	BigDoorjs		메인페이지 이전 대문페이지 구성
	css		체크리스트, 공지, 로그인페이지 로그인창에 사용될 css 폴더
	css_checkList		체크리스트에 사용될 css 폴더
	css_mypage		마이페이지의 카드형식 css 폴더
	fonts		Petriots 페이지에 사용될 폰트 폴더
	images		배너사진에 사용될 이미지들이 들어있는 폴더
	js		부트스트랩과 관련된 js가 들어있는 폴더
	vendor_mypage		부트스트랩에 관련된 css, js와 관련 js 폴더
	WEB-INF	views	Petriots 페이지가 들어있는 JSP가 담겨있는 폴더
Main	com.teamp .petriots.ccl_d_r .controller	ccl_d_rController.java	체크리스트 기본값 테이블 컨트롤러 구성
	com.teamp .petriots.ccl_d_r .dao	ICcl_d_rDAO.java	체크리스트 기본값 테이블 인터페이스 구성
	com.teamp .petriots.ccl_d_r .dto	Ccl_d_rDTO.java	체크리스트 기본값 테이블 dto 구성
	com.teamp .petriots.ccl_d_r .mapper	Ccl_d_rDAO.xml	체크리스트 기본값 테이블 dao.xml 구성
	com.teamp .petriots .ccl_request .controller	Ccl_requestController .java	요청 체크리스트 항목 테이블 컨트롤러 구성
	com.teamp .petriots .ccl_request .dao	ICcl_requestDAO .java	요청 체크리스트 항목 테이블 인터페이스 구성
	com.teamp .petriots .ccl_request .dto	Ccl_requestDTO .java	요청 체크리스트 항목 테이블 dto 구성
	com.teamp .petriots .ccl_request .mapper	Ccl_requestDAO .xml	요청 체크리스트 항목 테이블 dao.xml 구성

Depth1	Depth2	Depth3	비고
	com.teamp .petriots .entrance .controller	EntranceController .java	대문페이지, 메인페이지 이동을 위한 컨트롤러
	com.teamp .petriots.i_q_b .controller	I_q_bController.java	지식시험문제 테이블 컨트롤러 구성
	com.teamp .petriots.i_q_b .dao	I_q_bDAO.java	지식시험문제 테이블 인터페이스 구성
	com.teamp .petriots.i_q_b .dto	I_q_bDTO.java	지식시험문제 테이블 dto 구성
	com.teamp .petriots.i_q_b .mapper	I_q_bDAO.xml	지식시험문제 테이블 dao.xml 구성
	com.teamp .petriots.p_q_b .controller	P_q_bController.java	인성검사 테이블 컨트롤러 구성
	com.teamp .petriots.p_q_b .dao	IP_q_bDAO.java	인성검사 테이블 인터페이스 구성
	com.teamp .petriots.p_q_b .dto	P_q_bDTO.java	인성검사 테이블 dto 구성
	com.teamp .petriots.p_q_b .mapper	P_q_bDAO.xml	인성검사 테이블 dao.xml 구성

Depth1	Depth2	Depth3	비고
	com.teamp .petriots .pet_register .controller	Pet_registerController .java	반려동물등록 테이블 컨트롤러 구성
	com.teamp .petriots .pet_register .dao	Pet_registerDAO.java	반려동물등록 테이블 인터페이스 구성
	com.teamp .petriots .pet_register .dto	Pet_registerDTO.java	반려동물등록 테이블 dto 구성
	com.teamp .petriots .pet_register .mapper	Pet_registerDAO.xml	반려동물등록 테이블 dao.xml 구성
	com.teamp .petriots.question .controller	QuestionController .java	질문 테이블 컨트롤러 구성
	com.teamp .petriots.question .dao	IQuestionDAO.java	질문 테이블 인터페이스 구성
	com.teamp .petriots.question .dto	QuestionDTO.java	질문 테이블 dto 구성
	com.teamp .petriots.question .mapper	question.xml	질문 테이블 dao.xml 구성
	com.teamp .petriots .reg_my_ccl.dto	Reg_my_cclDTO.java	체크리스트번호, 동물번호, 시작시간 끝시간, 할 일 변수 getter / setter 구성
	com.teamp .petriots.report .controller	ReportController.java	신고 테이블 컨트롤러 구성
	com.teamp .petriots.report .dao	IReportDAO.java	신고 테이블 인터페이스 구성

Depth1	Depth2	Depth3	비고
	com.teamp .petriots.report .dto	ReportDTO.java	신고 테이블 dto 구성
	com.teamp .petriots.report .mapper	ReportDAO.xml	신고 테이블 dao.xml 구성
	com.teamp .petriots.signup .controller	SignupController.java	회원 등록 테이블 컨트롤러 구성
	com.teamp .petriots.signup .dao	ISignupDAO.java	회원 등록 테이블 인터페이스 구성
	com.teamp .petriots.signup .dto	SignupDTO.java	회원 등록 테이블 dto 구성
	com.teamp .petriots.signup .login.controller	MemberServlet.java	회원 등록 테이블 Servlet 구성
	com.teamp .petriots.signup .login.dao	MemberDAO.java	회원 등록 테이블 MemberDAO 구성
	com.teamp .petriots.signup .login.dto	MemberDTO.java SessionInfo.java	회원 등록 테이블 MemberDTO, SessionInfo로 세션 유지를 위한 getter / setter 구성
	com.teamp .petriots.signup .login.util	DBCPConn.java	데이터베이스 연결을 위한 dbcp 구성
	com.teamp .petriots.signup .mapper	SignupDAO.xml	회원 등록 테이블 dao.xml 구성

Depth1	Depth2	Depth3	비고
	com.teamp .petriots.util	MyUtil.java	게시판이 페이징 처리를 위한 구성
	com.teamp .petriots.withdrawal .controller	WithdrawalController .java	회원 탈퇴 테이블 컨트롤러 구성
	com.teamp .petriots.withdrawal .dao	IWithdrawalDAO.java	회원 탈퇴 테이블 인터페이스 구성
	com.teamp .petriots.withdrawal .dto	WithdrawalDTO.java	회원 탈퇴 테이블 dto 구성
	com.teamp .petriots.withdrawal .mapper	WithdrawalDAO.xml	회원 탈퇴 테이블 dao.xml 구성
	com.teamp .petriots .entrance .controller	EntranceController .java	대문페이지, 메인페이지 이동을 위한 컨트롤러
Petsiter	com.teamp .petriots.image	ImageDAO.java ImageServlet.java	반려동물 이미지와 회원 이미지 구성
	com.teamp .petriots .pst_profile .controller	Pst_profileController .java	펫시터 프로필 테이블 컨트롤러 구성
	com.teamp .petriots .pst_profile .dao	IPst_profileDAO.java	펫시터 프로필 테이블 인터페이스 구성
	com.teamp .petriots .pst_profile .dto	Pst_profileDTO.java	펫시터 프로필 테이블 dto 구성

Depth1	Depth2	Depth3	비고
	com.teamp .petriots .pst_profile .mapper	Pst_profileDAO.xml	펫시터 프로파일 테이블 dao.xml 구성
	com.teamp .petriots.pst_shift .controller	Pst_shiftController .java	펫시터 전환 테이블 컨트롤러 구성
	com.teamp .petriots.pst_shift .dao	IPst_shiftDAO.java	펫시터 전환 테이블 인터페이스 구성
	com.teamp .petriots.pst_shift .dao.mapper	Pst_shiftDAO.xml	펫시터 전환 테이블 dao.mapper 구성
	com.teamp .petriots.pst_shift .dto	Conduct_pDTO.java CutlineDTO.java Marking_pDTO.java P_q_bDTO.java pass_pDTO.java pass_tDTO.java Pst_shiftDTO.java Solve_pDTO.java	펫시터 전환 dto로 전환하기 위한 변수 사용을 위해 커트라인 인성문제 점수 구성의 dto 구성
	com.teamp .petriots .sit_approve .controller	IPst_reserveController .java	위탁 시작 테이블 컨트롤러 구성
	com.teamp .petriots .sit_approve .dao	IPst_reserveDAO.java	위탁 시작 테이블 인터페이스 구성
	com.teamp .petriots .sit_approve .dto	Pst_reserveDTO.java	위탁 시작 테이블 dto 구성
	com.teamp .petriots .sit_approve .mapper	Pst_reserveDAO.xml	위탁 시작 테이블 dao.xml 구성

Depth1	Depth2	Depth3	비고
Admin	com.teamp .petriots .cutline .controller	cutlineController .java	커트라인 테이블 컨트롤러 구성
	com.teamp .petriots .cutline .dao	ICutlineControllerDAO .java	커트라인 테이블 인터페이스 구성
	com.teamp .petriots .cutline .dto	CutlineDTO .java	커트라인 테이블 dto 구성
	com.teamp .petriots .cutline .mapper	CutlineDAO .xml	커트라인 테이블 dao.xml 구성
	com.teamp .petriots.mtom_q .controller	Mtom_qController .java	1대1문의 테이블 컨트롤러 구성
	com.teamp .petriots.mtom_q .dao	IMtom_qDAO.java	1대1문의 테이블 인터페이스 구성
	com.teamp .petriots.mtom_q .dto	Mtom_qDTO.java	1대1문의 테이블 dto 구성
	com.teamp .petriots.mtom_q .mapper	Mtom_qDAO.xml	1대1문의 테이블 dao.xml 구성
	com.teamp .petriots.notice .controller	NoticeController.java	공지사항 테이블 컨트롤러 구성
	com.teamp .petriots.notice .dao	INoticeDAO.java	공지사항 테이블 인터페이스 구성

Depth1	Depth2	Depth3	비고
	com.teamp .petriots.notice .dto	NoticeDTO.java	공지사항 테이블 dto 구성
	com.teamp .petriots.notice .mapper	notice.xml	공지사항 테이블 dao.xml 구성
	com.teamp .petriots .register_admin .controller	register_admin Controller.java	관리자 테이블 컨트롤러 구성
	com.teamp .petriots .register_admin .dao	IRegister_adminDAO .java	관리자 테이블 인터페이스 구성
	com.teamp .petriots .register_admin .dto	Register_adminDTO .java Register_admin SessionInfo.java	관리자 계정 세션유지를 위한 dto와 관리자 테이블 변수 getter / setter 구성을 위한 dto 구성
	com.teamp .petriots .register_admin .mapper	Register_adminDAO .xml	관리자 테이블 dao.xml 구성

**매음말
맛음말**

Location-Based WebApplication

이은채

<제목 : 노력하는 시간들이 모여 이뤄낸 결실>

한 달 남짓한 프로젝트 기간 동안 더 많은 기능들을 구현하지 못해 매우 아쉽지만 그에 반해 더욱더 보람찬 시간들이었다.

프로젝트 시작 전, 각자 조원들의 의견을 모아 아이디어를 구상하였다.

초반엔 자유롭게 편곡 할 수 있는 음악 웹 어플리케이션을 구성하였으나, 강사님의 컨펌을 통해 6개월의 과정을 배우면서 배웠던 내용을 최대한 많이 녹여낼 수 있는 사이트로 의견을 모아 최종적으로 PETRIOTS (펫시터 위치기반 서비스)가 탄생하게 되었다.

아이디어가 확정 된 후 요구사항과 그에 맞는 세부 정책들을 구상하였다.

처음엔 요구분석 기술과 정책 사항에 대해 기능들만 나열하고 반대 상황이나 사용자 측면에서 사이트를 이용하며 나올 수 있는 여러 가지 상황을 생각을 하지 못했던 부분들이 많아 다른 조에 비해 플로우 차트 작성을 늦게 시작하게 되었지만 그만큼 꼼꼼하게 아이디어를 구성할 수 있어서 플로우 차트는 한 번 만에 통과하여 ERD 논리설계를 바로 들어 갈 수 있었다.

DB 설계를 하면서 많은 어려움이 있었다.

페이지 특성상 예약과, 유저와 펫시터 간의 주고받는 체크리스트와, 시험 출제 때문에 테이블 관계도를 설정 하는 것에 있어 시간소요가 많이 들었다.

특히 시험문제 출제 같은 경우에는 문제 하나당 컬럼 하나로 표현하고 각각의 컬럼 들이 시험 테이블을 참조받아 구현해야 했기 때문에 많이 까다로웠지만 좋은 경험이 될 수 있다고 생각한다.

우열곡절 끝에 ERD를 통과하고 DB를 설계하고 오류를 고칠 동안 팀원들과 틈틈이 뷰 페이지를 작하면서 CSS와 부트스트랩을 자유롭게 이용할 수 있게 되었다.

초반에는 조그마한 검색필터 하나를 만드는 데에도 몇 시간이 소요되었고, 조원들의 뷰 페이지와 합칠 때,

CSS 충돌문제로 디자인이 어긋나는 경우가 발생하였다.

하지만, 해당부분은 <div display="flex" 속성을 통해 <div>의 배치관계를 이해하며 CSS의 기능들을 많이 공부 할 수 있었다.

수많은 뷰페이지를 제작하고 조원들과 협업하여 부족한 부분은 서로 도와주며 팀워크를 쌓을 수 있었다.

Spring MVC 구조를 통해 Model, View, Controller 순서를 꼼꼼히 재 학습 할 수 있었다.

Mybatis를 이용하여 Annotation 주로 사용하는 @RequestMapping과, @Autowired, @Controller 와 Mapping하여 뷰에서 action처리 부분을 진행 하였다.

뿐만 아니라, 많은 뷰 페이지를 제작하면서, CSS와 부트스트랩을 자유롭게 이용할 수 있게 되었다.

백엔드단 과 프론트 앤드단을 병행하며 프로젝트를 진행하여 보다 넓은 범위로 능력 향상되었다.

조원들에 비해 학습이 늦어 코드단 구성은 다른 조원에 비해 부족하게 구성하였지만, 자신 있는 부분은 조원들과 공유하며 실력을 쌓고 배움을 터득하였다.

초반에 기획했던 내용과는 달리 부분적으로 기능들(체크리스트 횡수제한 반려, 핫라인, 쪽지 기능) 을 전부 수행하지 못하였지만, 프로젝트가 끝나도 조원들과 각자 시간을 맞춰 초반에 기획 했던 것 보다 더욱더 전문적 있게 구성하여 완벽하게 프로젝트를 끝낼 예정이다.