04/25/2017

## Final Project - Animal Rescue Center System

## Qing Mu

## Topic

Stray Animal Rescue and Adoption Center Management System

## Background

Nowadays, more and more people keep pets such as dogs and cats. However, at the same time, we can see more stray animals in the street. In order to protect these abandoned animals, rescue centers have been established, which need a system to record and manage the animals.

## Brief Summary

**Development tools:**

* STS
* MySQL Database

**The techniques used in the project:**

* Spring MVC
* Hibernate (Annotation, Entities mapping, Queries, Criteria)
* Filter
* JSTL
* Bootstrap

## Functionality Summary

**All Users:**

* Sign up as adopters
* Login their accounts
* Log out

**Admin Role:**

* See statistic reports, including total quantity of the animals rescued, percentage by type, percentage by gender, and so on.

**Registrator Role:**

* Browse all the rescue records
* Register new animals and record their information, including type, gender, color and photo

**Veterinarian Role:**

* Browse all the rescue records
* Treat animals that haven’t been treated yet
* Once the animals are treated, they are then automatically added into an adoption list, which can been seen by adopters

**Adoption Supervisor Role:**

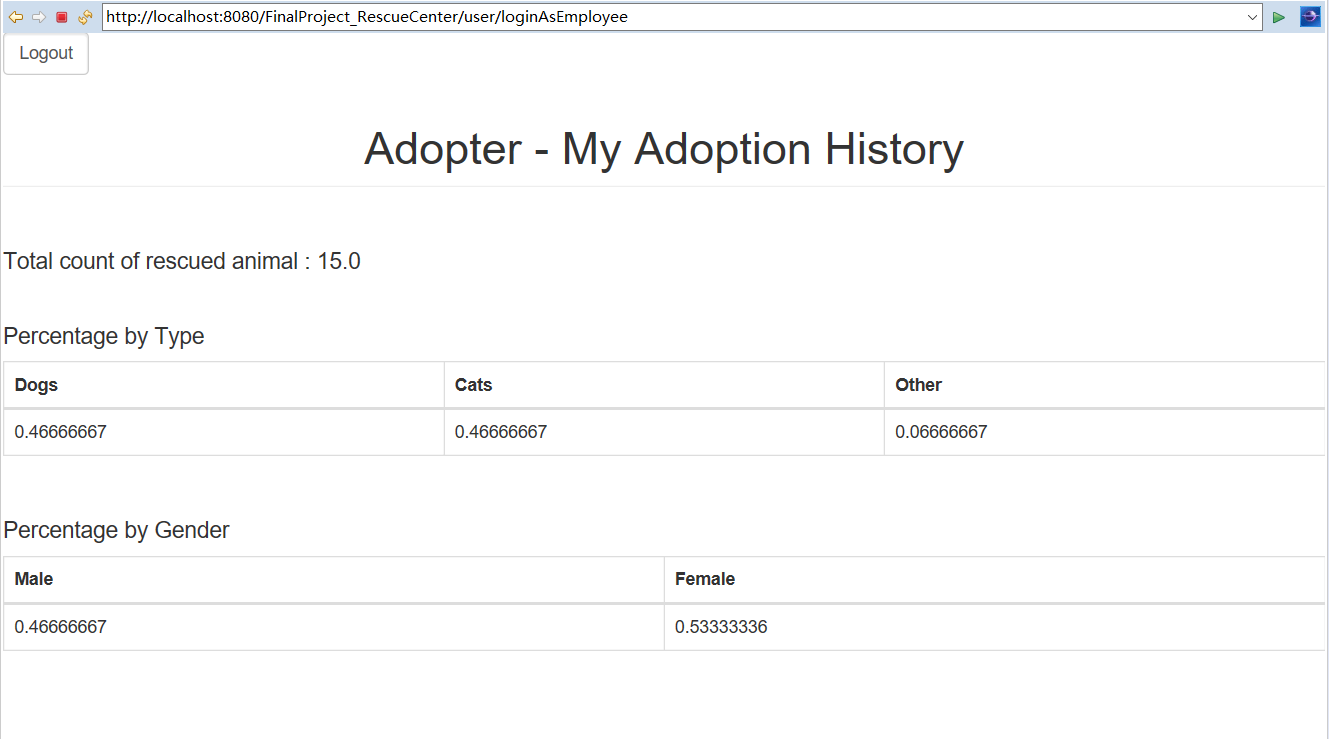
* Browse all the pending adoption applications
* See details of an application and evaluate whether the adopter is certificated.

**Adopter Role:**

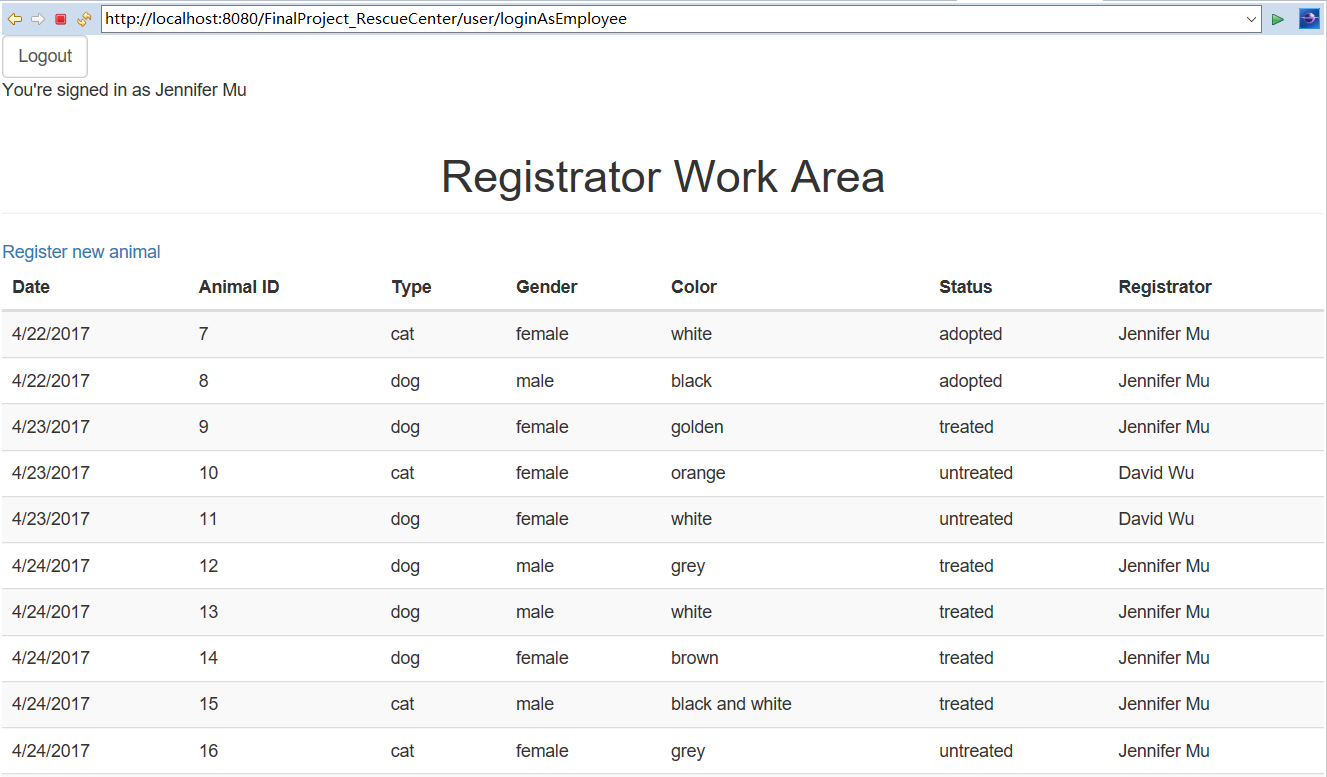
* Browse and edit their information
* Browse their own adoption histories
* Browse adoption list and see the detail information of an animal
* Apply for adopting an animal

## Screenshots of key pages

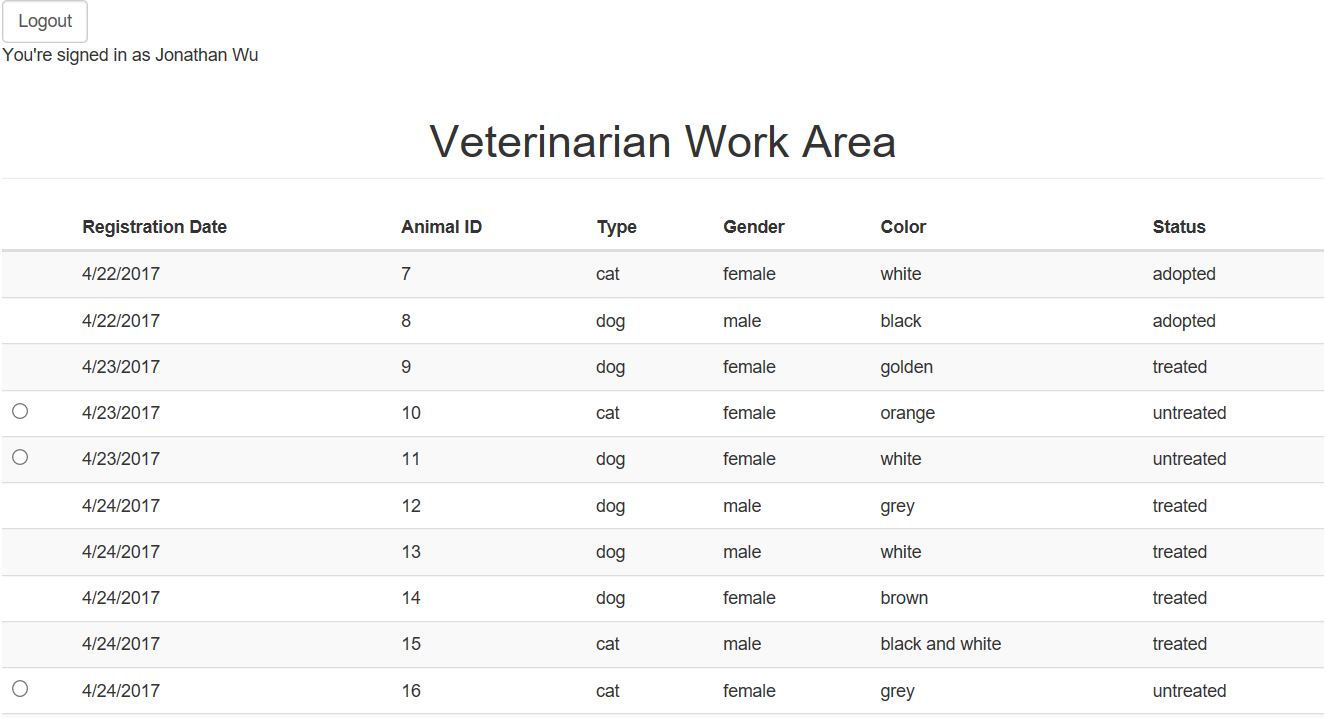
**Admin Role - Statistics Report**



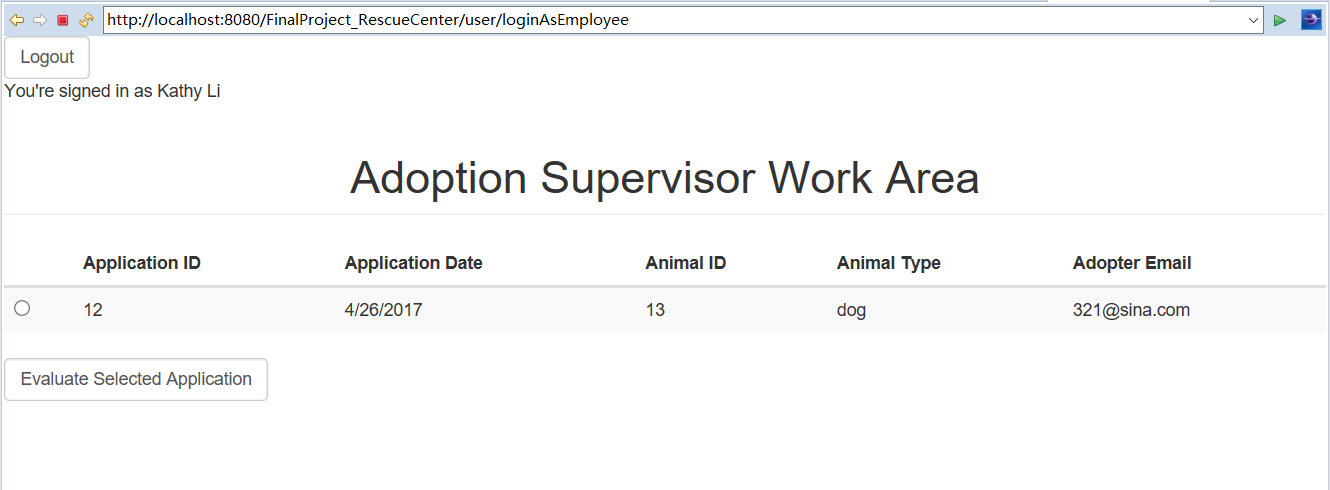
**Registrator Role - Dashboard ( browse all rescue records )**



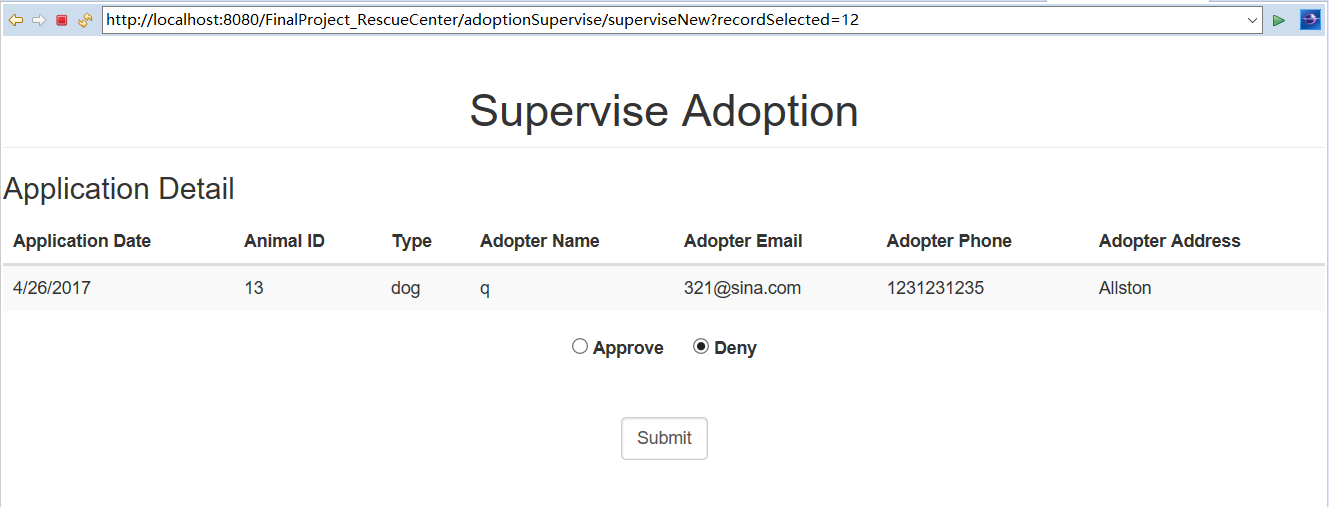
**Veterinarian Role - Dashboard ( browse all rescue records )**



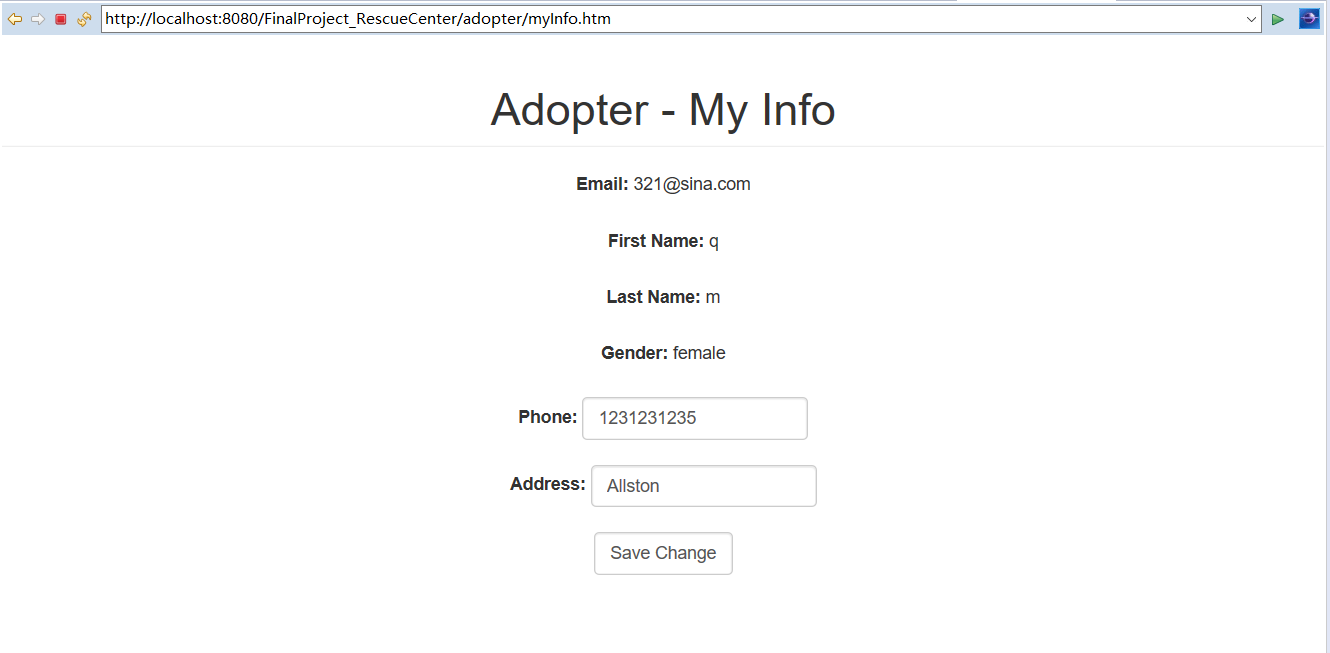
**Adoption Supervise Role - Dashboard ( browse all adoption applications )**



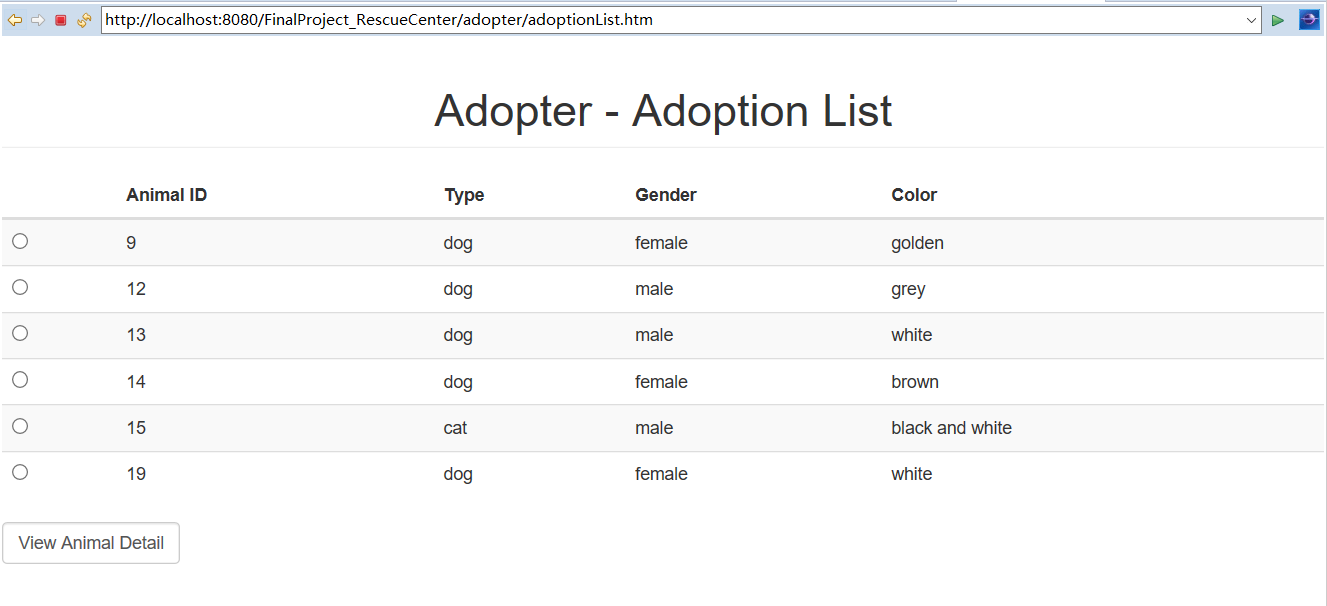
**Adoption Supervise Role - View Application Detail**



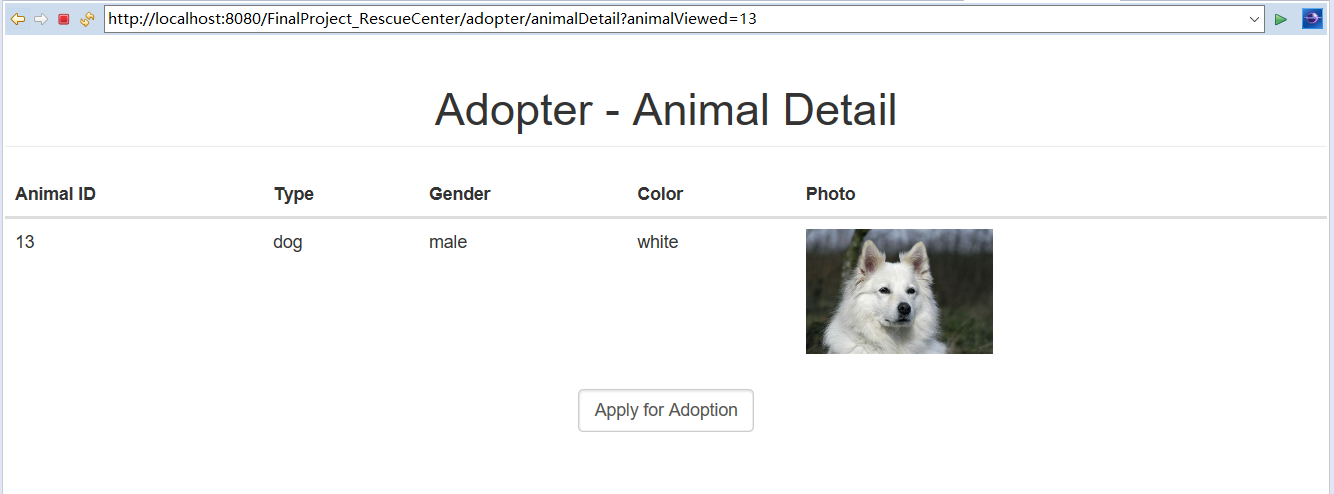
**Adopter Role - My Information**



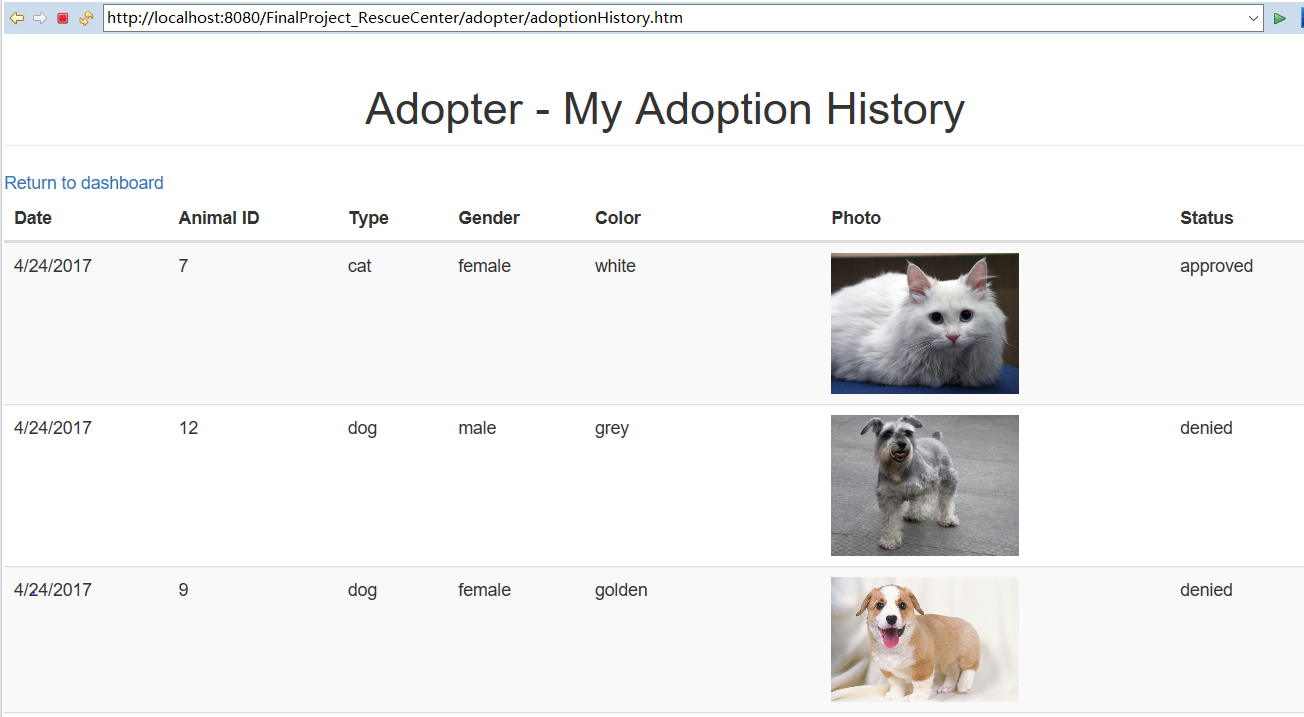
**Adopter Role - Adoption List**



**Adopter Role - Animal Detail**



**Adopter Role - Adoption History**



## Appendix: Controllers

### 【UserController.java】

**package** com.qing.controller;

**import** java.util.List;

**import** java.util.Locale;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** javax.servlet.http.HttpSession;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.ui.Model;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.RequestParam;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.qing.dao.AdoptionRecordDAO;

**import** com.qing.dao.AnimalDAO;

**import** com.qing.dao.RescueRecordDAO;

**import** com.qing.dao.TreatmentRecordDAO;

**import** com.qing.dao.UserDAO;

**import** com.qing.pojo.Adopter;

**import** com.qing.pojo.AdopterUserAccount;

**import** com.qing.pojo.AdoptionRecord;

**import** com.qing.pojo.Employee;

**import** com.qing.pojo.EmployeeUserAccount;

**import** com.qing.pojo.RescueRecord;

@Controller

**public** **class** UserController {

@Autowired

@Qualifier("userDAO")

UserDAO userDAO;

@Autowired

@Qualifier("animalDAO")

AnimalDAO animalDAO;

@Autowired

@Qualifier("rescueRecordDAO")

RescueRecordDAO rescueRecordDAO;

@Autowired

@Qualifier("treatmentRecordDAO")

TreatmentRecordDAO treatmentRecordDAO;

@Autowired

@Qualifier("adoptionRecordDAO")

AdoptionRecordDAO adoptionRecordDAO;

@RequestMapping(value="/user/logout", method=RequestMethod.***GET***)

**public** ModelAndView getLogout( HttpServletRequest request ) **throws** Exception{

HttpSession session = (HttpSession) request.getSession();

session.invalidate();

**return** **new** ModelAndView("user\_loginAsAdopter");

}

@RequestMapping(value="/user/signup",method = RequestMethod.***GET***)

**public** ModelAndView getSignUp() **throws** Exception{

**return** **new** ModelAndView("user\_signup");

}

@RequestMapping(value="/user/signup",method = RequestMethod.***POST***)

**public** ModelAndView postSignUp( HttpServletRequest request,

@RequestParam("email") String email,

@RequestParam("password") String password,

@RequestParam("password2") String password2,

@RequestParam("lastname") String lastname,

@RequestParam("firstname") String firstname,

@RequestParam("gender") String gender,

@RequestParam("address") String address,

@RequestParam("phone") String phone)

**throws** Exception{

HttpSession session = (HttpSession) request.getSession();

**try**{

**if**(userDAO.isAdopterEmailValid(email)) {

**if**(!password.equals(password2)) {

**return** **new** ModelAndView("user\_passwordDifferent");

}

**else** {

Adopter adopter = userDAO.addAdopter(email, password, lastname, firstname, gender, address, phone);

session.setAttribute("user", adopter);

session.setAttribute("userType", "adopter");

**return** **new** ModelAndView("adopter\_dashboard");

}

}

**else** {

**return** **new** ModelAndView("user\_emailInvalid");

}

}

**catch** (Exception e) {

System.***out***.println("UserController\_postSignUp: " + e);

}

**return** **new** ModelAndView("error");

}

@RequestMapping(value="/user/loginAsAdopter",method = RequestMethod.***GET***)

**public** ModelAndView getLoginAsAdopter(){

ModelAndView mv = **new** ModelAndView();

**return** **new** ModelAndView("user\_loginAsAdopter");

}

@RequestMapping(value="/user/loginAsEmployee", method = RequestMethod.***GET***)

**public** ModelAndView getLoginAsEmployee(){

ModelAndView mv = **new** ModelAndView();

**return** **new** ModelAndView("user\_loginAsEmployee");

}

@RequestMapping(value="/user/loginAsEmployee", method = RequestMethod.***POST***)

**public** ModelAndView postLoginEmployee( HttpServletRequest request,

@RequestParam("employeeid") **long** id,

@RequestParam("password") String password )

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

EmployeeUserAccount eua = userDAO.getEmployeeUserAccount(id, password);

Employee e = userDAO.getEmployee(eua);

System.***out***.println(e.getName());

**if**(e != **null**) {

session.setAttribute("user", e);

**if**(e.getRole().equals("registrator")) {

session.setAttribute("userType", "registrator");

List<RescueRecord> recordList = rescueRecordDAO.getRescueRecordList();

**for**(RescueRecord rr: recordList) {

rr.setAnimal(animalDAO.getAnimalById(rr.getAnimalid()));

rr.setRegistrator(userDAO.getEmployeeByID(rr.getRegistratorid()));

}

session.setAttribute("rescueRecordList", recordList);

mv.setViewName("registrator\_workarea");

mv.addObject("registrator", e);

**boolean** isEmpty = **false**;

**if**( recordList.size() == 0 ) {

isEmpty = **true**;

}

mv.addObject("isEmpty", isEmpty);

mv.addObject("message","No content.");

**return** mv;

}

**else** **if**(e.getRole().equals("veterinarian")) {

session.setAttribute("userType", "veterinarian");

List<RescueRecord> recordList = rescueRecordDAO.getRescueRecordList();

**for**(RescueRecord rr: recordList) {

rr.setAnimal(animalDAO.getAnimalById(rr.getAnimalid()));

rr.setRegistrator(userDAO.getEmployeeByID(rr.getRegistratorid()));

}

session.setAttribute("rescueRecordList", recordList);

mv.setViewName("veterinarian\_workarea");

**boolean** isEmpty = **false**;

**if**( recordList.size() == 0 ) {

isEmpty = **true**;

}

mv.addObject("isEmpty", isEmpty);

mv.addObject("message","No content.");

**return** mv;

}

**else** **if**(e.getRole().equals("adoption")) {

session.setAttribute("userType", "adoption");

mv.setViewName("adoption\_workarea");

List<AdoptionRecord> recordList = adoptionRecordDAO.getAdoptionRecordListPending();

**for**(AdoptionRecord r: recordList) {

r.setAnimal(animalDAO.getAnimalById(r.getAnimalId()));

r.setAdopter(userDAO.getAdopterByEmail(r.getAdopterEmail()));

}

session.setAttribute("records", recordList);

**boolean** isEmpty = **false**;

**if**( recordList.size() == 0 ) {

isEmpty = **true**;

}

mv.addObject("isEmpty", isEmpty);

mv.addObject("message","No content.");

**return** mv;

}

**else** **if**(e.getRole().equals("admin")) {

session.setAttribute("userType", "admin");

mv.setViewName("admin\_report");

mv.addObject("total", animalDAO.getTotalCount());

mv.addObject("dogPercentage", animalDAO.getDogPercentage());

mv.addObject("catPercentage", animalDAO.getCatPercentage());

mv.addObject("otherPercentage", animalDAO.getOtherPercentage());

mv.addObject("malePercentage", animalDAO.getMalePercentage());

mv.addObject("femalePercentage", animalDAO.getFemalePercentage());

**return** mv;

}

**else** {

System.***out***.println("Role exception.");

**return** **new** ModelAndView("user\_userNotFound");

}

}

}

**catch** (Exception e) {

System.***out***.println(e);

mv.setViewName("user\_userNotFound");

}

**return** mv;

}

@RequestMapping(value="/user/loginAsAdopter", method=RequestMethod.***POST***)

**public** ModelAndView postLoginAdopter( HttpServletRequest request,

@RequestParam("email") String email,

@RequestParam("password") String password )

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

AdopterUserAccount aua = userDAO.getAdopterUserAccount(email, password);

Adopter a = userDAO.getAdopter(aua);

**if**(a != **null**) {

System.***out***.println("a is not null.");

mv.setViewName("adopter\_dashboard");

session.setAttribute("user", a);

session.setAttribute("userType", "adopter");

mv.addObject("adopter", a);

}

**else** {

System.***out***.println("User not found.");

**return** **new** ModelAndView("user\_userNotFound");

}

}

**catch** (Exception e) {

System.***out***.println(e);

mv.setViewName("user\_userNotFound");

}

**return** mv;

}

@RequestMapping(value="/",method = RequestMethod.***GET***)

**public** String home(Locale locale, Model model){

**return** "login";

}

}

### 【AdminController.java】

**package** com.qing.controller;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.qing.dao.AnimalDAO;

@Controller

**public** **class** AdminController {

@Autowired

@Qualifier("animalDAO")

AnimalDAO animalDAO;

@RequestMapping(value="/admin/report",method = RequestMethod.***GET***)

**public** ModelAndView getadminReport( ) **throws** Exception{

ModelAndView mv = **new** ModelAndView();

**try** {

mv.addObject("total", animalDAO.getTotalCount());

mv.addObject("dogPercentage", animalDAO.getDogPercentage());

mv.addObject("catPercentage", animalDAO.getCatPercentage());

mv.addObject("otherPercentage", animalDAO.getOtherPercentage());

mv.setViewName("adopter\_dashboard");

}

**catch**(Exception e) {

System.***out***.println(e);

mv.setViewName("error");

}

**return** mv;

}

}

### 【RegisterController.java】

**package** com.qing.controller;

**import** java.io.File;

**import** java.util.Calendar;

**import** java.util.List;

**import** java.util.Locale;

**import** java.io.IOException;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpSession;

**import** javax.servlet.ServletContext;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.ModelAttribute;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.RequestParam;

**import** org.springframework.web.multipart.commons.CommonsMultipartFile;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.qing.dao.AnimalDAO;

**import** com.qing.dao.RescueRecordDAO;

**import** com.qing.dao.UserDAO;

**import** com.qing.exception.RescueRecordException;

**import** com.qing.pojo.Animal;

**import** com.qing.pojo.Employee;

**import** com.qing.pojo.RescueRecord;

@Controller

**public** **class** RegisterController {

@Autowired

@Qualifier("animalDAO")

AnimalDAO animalDAO;

@Autowired

@Qualifier("userDAO")

UserDAO userDAO;

@Autowired

@Qualifier("rescueRecordDAO")

RescueRecordDAO rescueRecordDAO;

@Autowired

ServletContext servletContext;

@RequestMapping(value="/registration/registrator\_workarea",method = RequestMethod.***GET***)

**public** ModelAndView getRegisterWorkarea() **throws** RescueRecordException{

ModelAndView mv = **new** ModelAndView();

**try** {

System.***out***.println("Inside getRegisterWorkarea()");

mv.setViewName("registrator\_workarea");

mv.addObject("recordList", rescueRecordDAO.getRescueRecordList());

}

**catch**(Exception e) {

System.***out***.println("getRegisterWorkarea() : " + e);

}

**return** mv;

}

@RequestMapping(value="/registration/newAnimal",method = RequestMethod.***GET***)

**public** ModelAndView getRegisterNew() **throws** RescueRecordException{

ModelAndView mv = **new** ModelAndView();

**try** {

mv.setViewName("registrator\_addAnimal");

mv.addObject("recordList", rescueRecordDAO.getRescueRecordList());

}

**catch**(Exception e) {

System.***out***.println("getRegisterNew() : " + e);

}

**return** mv;

}

@RequestMapping(value="/registration/newAnimal", method=RequestMethod.***POST***)

**public** ModelAndView postRegisterNew( HttpServletRequest request,

@RequestParam("type") String type,

@RequestParam("gender") String gender,

@RequestParam("color") String color,

@ModelAttribute("model") Animal model)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

**if** (model.getFile() != "" || model.getFile() != **null**) {

File directory;

String check = File.***separator***; // Checking if system is linux

// based or windows based by

// checking seprator used.

String path = **null**;

**if** (check.equalsIgnoreCase("\\")) {

path = servletContext.getRealPath("").replace("build\\", ""); // gives real path as Lab9/build/web/

System.***out***.println("2 - " + path); // so we need to replace build in the path

}

System.***out***.println("3 - " + path);

directory = **new** File(path + "\\" + model.getPhoto());

System.***out***.println("4 - " + directory);

**boolean** temp = directory.exists();

**if** (!temp) {

temp = directory.mkdir();

}

**if** (temp) {

// We need to transfer to a file

CommonsMultipartFile photoInMemory = model.getPhoto();

String fileName = photoInMemory.getOriginalFilename();

// could generate file names as well

File localFile = **new** File(directory.getPath(), fileName);

// move the file from memory to the file

photoInMemory.transferTo(localFile);

model.setFile(localFile.getPath());

System.***out***.println("File is stored at " + localFile.getPath());

}

}

model.setType(type);

model.setGender(gender);

model.setColor(color);

model.setStatus("untreated");

animalDAO.addAnimal(model);

RescueRecord rr = **new** RescueRecord();

Employee e = (Employee) session.getAttribute("user");

rr.setAnimal(model);

rr.setAnimalid(model.getAnimalId());

rr.setRegistrator(e);

rr.setRegistratorid(e.getEmployeeId());

Calendar c = Calendar.*getInstance*();

**int** year = c.get(Calendar.***YEAR***);

**int** month = c.get(Calendar.***MONTH***);

**int** date = c.get(Calendar.***DATE***);

rr.setDate(month+1, date, year);

rescueRecordDAO.addRescueRecord(rr);

List<RescueRecord> recordList = rescueRecordDAO.getRescueRecordList();

**for**(RescueRecord r: recordList) {

r.setAnimal(animalDAO.getAnimalById(r.getAnimalid()));

r.setRegistrator(userDAO.getEmployeeByID(r.getRegistratorid()));

}

session.setAttribute("rescueRecordList", recordList);

mv.setViewName("registrator\_workarea");

**return** mv;

}

**catch** (Exception e) {

System.***out***.println(e + ": " + e.getMessage() + ": " + e.getCause());

}

**return** mv;

}

}

### 【TreatmentController.java】

**package** com.qing.controller;

**import** java.util.Calendar;

**import** java.util.List;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** javax.servlet.http.HttpSession;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.RequestParam;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.qing.dao.AnimalDAO;

**import** com.qing.dao.RescueRecordDAO;

**import** com.qing.dao.TreatmentRecordDAO;

**import** com.qing.dao.UserDAO;

**import** com.qing.pojo.Animal;

**import** com.qing.pojo.Employee;

**import** com.qing.pojo.RescueRecord;

**import** com.qing.pojo.TreatmentRecord;

@Controller

**public** **class** TreatmentController {

@Autowired

@Qualifier("animalDAO")

AnimalDAO animalDAO;

@Autowired

@Qualifier("userDAO")

UserDAO userDAO;

@Autowired

@Qualifier("rescueRecordDAO")

RescueRecordDAO rescueRecordDAO;

@Autowired

@Qualifier("treatmentRecordDAO")

TreatmentRecordDAO treatmentRecordDAO;

@RequestMapping(value="/veterinarian/newTreat",method = RequestMethod.***GET***)

**public** ModelAndView getTreat( HttpServletRequest request,

@RequestParam ("animalSelected") **long** animalid)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

mv.setViewName("veterinarian\_treat");

//List<Animal> animalList = animalDAO.getAnimalList();

Animal animal = animalDAO.getAnimalById(animalid);

System.***out***.println("getTreat: " + animal.getAnimalId());

session.setAttribute("animalTreated", animal);

mv.addObject("animalTreated", animal);

}

**catch**(Exception e) {

System.***out***.println(e);

}

**return** mv;

}

@RequestMapping(value="/veterinarian/newTreat", method=RequestMethod.***POST***)

**public** ModelAndView postTreat( HttpServletRequest request )

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

TreatmentRecord tr = **new** TreatmentRecord();

Employee e = (Employee) session.getAttribute("user");

Animal a = (Animal) session.getAttribute("animalTreated");

tr.setVeterinarian(e);

tr.setAnimal(a);

Calendar c = Calendar.*getInstance*();

**int** year = c.get(Calendar.***YEAR***);

**int** month = c.get(Calendar.***MONTH***);

**int** date = c.get(Calendar.***DATE***);

tr.setDate(month+1, date, year);

tr.setAnimalid(a.getAnimalId());

tr.setVeterinarianid(e.getEmployeeId());

animalDAO.updateAnimal(a.getAnimalId(), "treated");

treatmentRecordDAO.addTreatmentRecord(tr);

List<RescueRecord> recordList = rescueRecordDAO.getRescueRecordList();

**for**(RescueRecord rr: recordList) {

rr.setAnimal(animalDAO.getAnimalById(rr.getAnimalid()));

rr.setRegistrator(userDAO.getEmployeeByID(rr.getRegistratorid()));

}

session.setAttribute("rescueRecordList", recordList);

mv.setViewName("veterinarian\_workarea");

**return** mv;

}

**catch** (Exception e) {

System.***out***.println(e);

}

**return** mv;

}

}

### 【AdoptionSuperviseController.java】

**package** com.qing.controller;

**import** java.util.List;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpSession;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.RequestParam;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.qing.dao.AdoptionRecordDAO;

**import** com.qing.dao.AnimalDAO;

**import** com.qing.dao.UserDAO;

**import** com.qing.pojo.Adopter;

**import** com.qing.pojo.AdoptionRecord;

**import** com.qing.pojo.Animal;

@Controller

**public** **class** AdoptionSuperviseController {

@Autowired

@Qualifier("adoptionRecordDAO")

AdoptionRecordDAO adoptionDAO;

@Autowired

@Qualifier("animalDAO")

AnimalDAO animalDAO;

@Autowired

@Qualifier("userDAO")

UserDAO userDAO;

@Autowired

@Qualifier("adoptionRecordDAO")

AdoptionRecordDAO adoptionRecordDAO;

@RequestMapping(value="/adoptionSupervise/adoptionDashboard",method = RequestMethod.***GET***)

**public** ModelAndView getAdoptionDashboard( HttpServletRequest request )

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

mv.setViewName("adoption\_workarea");

List<AdoptionRecord> recordList = adoptionRecordDAO.getAdoptionRecordListPending();

**for**(AdoptionRecord r: recordList) {

r.setAnimal(animalDAO.getAnimalById(r.getAnimalId()));

r.setAdopter(userDAO.getAdopterByEmail(r.getAdopterEmail()));

}

session.setAttribute("records", recordList);

}

**catch**(Exception e){

System.***out***.println("AdoptionSuperviserController - getAdoptionDashboard");

mv.setViewName("ERROR");

}

**return** mv;

}

@RequestMapping(value="/adoptionSupervise/superviseNew", method = RequestMethod.***GET***)

**public** ModelAndView postAdoptionDashboard( HttpServletRequest request,

@RequestParam ("recordSelected") **long** recordid)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

mv.setViewName("adoption\_superviseNew");

AdoptionRecord record = (AdoptionRecord) adoptionDAO.getAdoptionRecordByID(recordid);

Animal animal = animalDAO.getAnimalById(record.getAnimalId());

Adopter adopter = (Adopter) userDAO.getAdopterByEmail(record.getAdopterEmail());

session.setAttribute("recordSelected", record);

session.setAttribute("animalSupervised", animal);

session.setAttribute("adopterSupervised", adopter);

}

**catch**(Exception e){

System.***out***.println("AdoptionSuperviserController - getAdoptionDashboard");

mv.setViewName("error");

}

**return** mv;

}

@RequestMapping(value="/adoptionSupervise/superviseNew", method=RequestMethod.***POST***)

**public** ModelAndView postAdoptionApproved( HttpServletRequest request,

@RequestParam ("statusSelected") String statusSelected)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

System.***out***.println("inside postAdoptionApproved");

**try**{

AdoptionRecord ar = (AdoptionRecord)session.getAttribute("recordSelected");

ar.setStatus(statusSelected);

adoptionDAO.updateAdoptionRecord(ar.getId(), statusSelected);

Animal animal = (Animal)session.getAttribute("animalSupervised");

**if**(statusSelected.equals("approved")) {

animal.setStatus("adopted");

animalDAO.updateAnimal(animal.getAnimalId(), "adopted");

}

**else** {

animal.setStatus("treated");

animalDAO.updateAnimal(animal.getAnimalId(), "treated");

}

mv.setViewName("adoption\_workarea");

session.setAttribute("records", adoptionDAO.getAdoptionRecordListPending());

**boolean** isEmpty = **false**;

**if**( adoptionDAO.getAdoptionRecordListPending().size() == 0 ) {

isEmpty = **true**;

}

mv.addObject("isEmpty", isEmpty);

**return** mv;

}

**catch** (Exception e) {

System.***out***.println(e);

mv.setViewName("error");

}

**return** mv;

}

}

### 【AdopterController.java】

**package** com.qing.controller;

**import** java.util.Calendar;

**import** java.util.List;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpSession;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.RequestParam;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.qing.dao.AdoptionRecordDAO;

**import** com.qing.dao.AnimalDAO;

**import** com.qing.dao.UserDAO;

**import** com.qing.pojo.Adopter;

**import** com.qing.pojo.AdoptionRecord;

**import** com.qing.pojo.Animal;

@Controller

**public** **class** AdopterController {

@Autowired

@Qualifier("animalDAO")

AnimalDAO animalDAO;

@Autowired

@Qualifier("adoptionRecordDAO")

AdoptionRecordDAO adoptionDAO;

@Autowired

@Qualifier("userDAO")

UserDAO userDAO;

@RequestMapping(value="/adopter/dashboard",method = RequestMethod.***GET***)

**public** ModelAndView getAdoptionDashboard( ) **throws** Exception{

**return** **new** ModelAndView("adopter\_dashboard");

}

@RequestMapping(value="/adopter/adoptionList",method = RequestMethod.***GET***)

**public** ModelAndView getAdoptionList( HttpServletRequest request) **throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

mv.setViewName("adopter\_adoptionlist");

mv.addObject("adoptionList", animalDAO.getAdopterAnimalAdoptionList());

session.setAttribute("adoptionList", animalDAO.getAdopterAnimalAdoptionList());

}

**catch**(Exception e){

System.***out***.println("AdopterController - getAdoptionList");

}

**return** mv;

}

@RequestMapping(value="/adopter/adoptionList", method=RequestMethod.***POST***)

**public** ModelAndView postAdoptionList( HttpServletRequest request,

@RequestParam ("animalSelected") String animalIndex)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

List<Animal> animalList = animalDAO.getAdopterAnimalAdoptionList();

Animal animal = animalList.get(Integer.*parseInt*(animalIndex));

session.setAttribute("animalSelected", animal);

mv.addObject("animalSelected", animal);

mv.setViewName("adopter\_animalDetail");

}

**catch** (Exception e) {

System.***out***.println("AdopterController - postAdoptionList");

}

**return** mv;

}

@RequestMapping(value="/adopter/animalDetail", method = RequestMethod.***GET***)

**public** ModelAndView getAnimalDetail( HttpServletRequest request,

@RequestParam ("animalViewed") **long** animalid)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

mv.setViewName("adopter\_animalDetail");

//List<Animal> animalList = animalDAO.getAnimalList();

Animal animal = animalDAO.getAnimalById(animalid);

System.***out***.println("getAnimalDetail: " + animal.getAnimalId());

session.setAttribute("animalViewed", animal);

mv.addObject("animalViewed", animal);

}

**catch**(Exception e) {

System.***out***.println(e);

mv.setViewName("error");

}

**return** mv;

}

@RequestMapping(value="/adopter/animalDetail", method = RequestMethod.***POST***)

**public** ModelAndView postAnimalDetail( HttpServletRequest request ) **throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

Adopter adopter = (Adopter) session.getAttribute("user");

Animal animal = (Animal) session.getAttribute("animalViewed");

AdoptionRecord record = **new** AdoptionRecord();

System.***out***.println(adopter.getEmail());

record.setAdopter(adopter);

record.setAdopterEmail(adopter.getEmail());

record.setAnimal(animal);

record.setAnimalId(animal.getAnimalId());

System.***out***.println(record.getAnimalId());

record.setStatus("pending");

record.getAnimal().setStatus("pending");

Calendar c = Calendar.*getInstance*();

**int** year = c.get(Calendar.***YEAR***);

**int** month = c.get(Calendar.***MONTH***);

**int** date = c.get(Calendar.***DATE***);

record.setDate(month+1, date, year);

adoptionDAO.addAdoptionRecord(record);

record.getAnimal().setStatus("treated");

animalDAO.updateAnimal(record.getAnimal().getAnimalId(), "pending");

mv.setViewName("adopter\_myAdoptionHistory");

List<AdoptionRecord> myAdoptions = adoptionDAO.getAdoptionRecordListByAdopter(adopter);

**for**(AdoptionRecord ar: myAdoptions) {

ar.setAdopter(adopter);

ar.setAnimal(animalDAO.getAnimalById(ar.getAnimalId()));

}

session.setAttribute("myAdoptions", myAdoptions);

session.setAttribute("adoptionList", animalDAO.getAdopterAnimalAdoptionList());

}

**catch**(Exception e){

System.***out***.println("AdopterController - getAdoptionList");

mv.setViewName("error");

}

**return** mv;

}

@RequestMapping(value="/adopter/myInfo",method = RequestMethod.***GET***)

**public** ModelAndView getMyInfo(HttpServletRequest request)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

mv.setViewName("adopter\_myInfo");

Adopter adopter = (Adopter) session.getAttribute("user");

mv.addObject("adopter", adopter);

}

**catch**(Exception e){

System.***out***.println("AdopterController - getMyInfo");

}

**return** mv;

}

@RequestMapping(value="/adopter/myInfo", method = RequestMethod.***POST***)

**public** ModelAndView postMyInfo( HttpServletRequest request,

@RequestParam ("phone") String phone,

@RequestParam ("address") String address)

**throws** Exception{

System.***out***.println("postMyInfo1");

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

List<Animal> animalList = animalDAO.getAdopterAnimalAdoptionList();

Adopter adopter = (Adopter) session.getAttribute("user");

adopter.setPhone(phone);

adopter.setAddress(address);

String email = adopter.getEmail();

System.***out***.println("postMyInfo2");

userDAO.updateAdopter(email, phone, address);

mv.setViewName("adopter\_dashboard");

}

**catch** (Exception e) {

System.***out***.println("AdopterController - postMyInfo");

mv.setViewName("error");

}

**return** mv;

}

@RequestMapping(value="/adopter/adoptionHistory",method = RequestMethod.***GET***)

**public** ModelAndView getMyHistory(HttpServletRequest request)

**throws** Exception{

ModelAndView mv = **new** ModelAndView();

HttpSession session = (HttpSession) request.getSession();

**try**{

mv.setViewName("adopter\_myAdoptionHistory");

Adopter adopter = (Adopter) session.getAttribute("user");

List<AdoptionRecord> myAdoptions = adoptionDAO.getAdoptionRecordListByAdopter(adopter);

**for**(AdoptionRecord ar: myAdoptions) {

ar.setAdopter(adopter);

ar.setAnimal(animalDAO.getAnimalById(ar.getAnimalId()));

}

session.setAttribute("myAdoptions", myAdoptions);

}

**catch**(Exception e){

System.***out***.println("AdopterController - getMyInfo");

mv.setViewName("error");

}

**return** mv;

}

}