



**本科毕业论文**

|  |
| --- |
| **宠物犬疾病诊断专家系统–Android客户端** |
|  |

**林锡鑫**

**201230560215**

|  |  |
| --- | --- |
| 指导教师 | **吴春胤 副教授** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学院名称 |  | **数学与信息学院** | 专业名称 |  | **信息管理与信息系统** |
| 论文提交日期 |  | 2016年4月15日 | 论文答辩日期 |  | 2016年5月 5 日 |

**摘 要**

虽然我国宠物犬的数量在上升，但对宠物犬的诊断治疗仍然以经验为主，国内专家系统在宠物犬医学领域的研究、发展和应用相对较少。但是近年来，随着我国经济的发展、计算机的普及，通过从事计算机研究的专家和兽医人员的共同努力，专家系统在兽医领域逐渐得到重视和应用。

本文基于Android应用程序开发技术以及Java Web技术开发了基于Android客户端的宠物犬疾病诊断专家系统。系统以实现核心功能为目的，提高用户体验为目标，采用了面向对象的思想对系统功能模块进行分析和设计。系统主要通过http请求的方式实现Android客户端和Web服务器之间的通讯交互。Web后台采用 Spring MVC 框架和MyBatis为基础的Web应用框架，为Android客户端提供丰富的接口，采用MySQL作为数据库。Android客户端的开发运用了一些开源组件如pulltorefresh，nineoldandroids等，增加了界面的美观性，提升了用户体验。

宠物犬疾病诊断专家系统实现了根据宠物犬的症状对宠物犬所患的疾病的诊断，为用户提供宠物犬疾病介绍和治疗措施，并且增加了多样化的功能如疾病文章推送，用户反馈等功能来提高用户体验，通过收集宠物犬的诊断记录和用户对诊断的反馈来提高专家系统诊断的准确性。

关键词：宠物 疾病诊断 专家系统 Android Web

**Expert System for Pet Dog Disease Diagnosis Based on Android**

Lin Xixin

(College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China)

**Abstract:** Although the number of pet dogs is on the rise, the treatment of pet dogs is still empirical. In recent years, with the Chinese economic development and popularity of computers, expert systems in the veterinary fields has attracted more attention and been widely used.

Expert system for pet dog disease diagnosis is developed as Android application based on Android application development technologies and Java Web technology. The system regards achieving the core functionality as purpose and aims at improving the user experience. The system function module is analyzed and designed with the idea of object oriented. The main way that system uses to realize the communication interaction between Android client and Web server is http request. Back-end system uses Spring MVC framework and MyBatis which based on Web application framework to provide a rich client interface for Android. And it uses MySQL as the database. The development of Android application uses some open source components such as pulltorefresh, nineoldandroids to increase the aesthetics of interface and enhance the user experience.

Expert system for pet dog disease diagnosis achieves diagnosing the disease which pet dogs suffer from according to the diagnosis of the symptoms and providing users with a presentation of disease and treatment. To improve the user experience, the system increase the variety of features, such as disease article pushing, user feedback and other functions. By collecting diagnostic record for pet dogs and user feedback, it improves the accuracy of diagnosis expert system.

**Key words:** pet dog disease diagnosis expert system Android Web

**目 录**

1 前言 1

1.1 研究背景及意义 1

1.2 研究现状 1

1.3 研究内容和方案 1

2 系统开发技术 2

2.1 Spring 2

2.1.1 控制反转 3

2.1.2 面向切面 3

2.2 Spring MVC框架 3

2.3 MyBatis 4

2.4 MySQL 4

2.5 Android系统 4

3 系统需求分析 5

3.1 涉众及系统用例 5

3.2 系统主要用例 6

3.2.1 登陆系统用例 7

3.2.2 忘记密码用例 8

3.2.3 疾病诊断用例 9

3.2.4 疾病诊断反馈用例 10

3.2.5 获取疾病文章推送用例 11

4 系统概要设计 11

4.1 系统功能和模块的划分 11

4.2 技术架构 13

4.2.1 Android客户端和Web服务器的通讯 13

4.2.2 图片存储技术 13

4.2.3 云推送 14

4.2.4 邮件服务端 14

4.3 数据库设计 15

4.3.1 E-R图设计 15

4.3.2 数据库表设计 18

5 系统详细设计 21

5.1 注册登录模块 21

5.1.1 用户登录 21

5.1.2 用户注册和信息修改 23

5.1.3 忘记密码 25

5.2 疾病诊断模块 27

5.2.1 疾病诊断流程 27

5.2.2 疾病诊断反馈 30

5.3 疾病文章推送模块 31

5.3.1 疾病文章推送流程 31

5.3.2 疾病文章列表获取 32

6 系统实现 33

6.1 用户登录 33

6.2 疾病诊断流程 34

6.2 疾病诊断反馈 35

7 结论与讨论 35

7.1 结论 35

7.2 讨论 36

参考文献 37

致谢 38

华南农业大学本科生毕业论文成绩评定表

1 前言

1.1 研究背景及意义

近年来，越来越多的人养宠物犬，目前由于在基层进行宠物犬疾病诊断的专业临床从业人员相对较少，宠物犬很难得到专业的治疗。在宠物犬出现症状的时候，人们会更偏向于自己通过互联网查询或者提问的方式去得到治疗方案。而网上的治疗方案各不相同，缺乏专业性和统一性。因此有必要开发可以诊断宠物犬疾病的专家系统。

针对国内对诊断宠物犬疾病的系统的迫切需求，本研究通过对宠物犬疾病诊断流程的模拟，开发出宠物犬疾病诊断专家系统，填补国内尚无宠物犬疾病诊断专家系统的空白，解决基层动物疾病诊断水平较低等问题。

专家系统在宠物犬疾病诊断中的应用是符合我国宠物犬行业发展状况和国际人工智能发展趋势的。通过专业的角度对宠物犬疾病诊断专家系统的研究，可以解决目前宠物犬增多，专业兽医不足的问题，为专业人员的知识积累也提供了更好的平台。

1.2 研究现状

专家系统已发展到众多领域，并产生了巨大的社会效益和经济效益。它实现了人工智能从理论研究走向实际应用，从一般推理策略探讨转向运用专门知识的重大突破，成为人工智能从实验室研究进入实用领域的一个里程碑。

我国宠物行业虽然已经发展了十多年,但是与国外发达国家相比还处于行业发展初期。由于宠物犬疾病种类多，诊断过程复杂，现阶段在互联网行业里面的产品，如乐宠等app，对宠物犬的关注点以社区，上门服务，电商为主，以宠物疾病为核心的应用相对较少。专家系统的使用在宠物犬疾病领域仍然有很大的发展空间。

1.3 研究内容和方案

专家系统是目前人工智能研究中最活跃、最有成效的领域之一（郑家三， 2002）。它的智能化主要表现为能够在特定的领域内模仿人类专家思维来求解复杂问题（王晓娟，2014）。本系统通过对宠物犬疾病诊断的过程进行调研，结合专家系统的功能，对诊断过程进行量化和建模，并增加用户登录注册，诊断反馈，疾病文章推送等功能来丰富客户端的功能。在系统实现方面，项目由Android手机客户端应用程序，Web后台服务器，数据库组成。对疾病进行诊断等业务逻辑主要写在后台当中，Android手机客户端主要是和用户进行交互，获取用户输入的数据，并将用户请求发送到后台，后台通过不同的业务逻辑处理前端发送来的请求爱，并根据业务逻辑从数据库中获取所需要的数据，返回给Android应用程序客户端。

系统还用到了百度云推送进行疾病文章的推送，使用七牛云存储进行图片的存储。

本文主要通过理论学习和系统开发的方法进行研究。本项目的实施方案可分为：

（1）调研分析。通过查阅相关资料，了解专家系统的作用的组成结构，并且了解宠物犬疾病诊断在国内的状况，通过对知识的整理设计出系统的总体构成。

（2）系统设计。根据大致的需求进行模块划分，和主要诊断流程的设计。

（3）系统实现。包括代码编写和代码测试和优化。

系统的开发平台如下：

1. 开发语言：Java；
2. 操作系统：Android

（3）集成开发环境：Eclipse Java EE、Android Studio

（4）数据库：MySQL 5.7；

（5）Web容器：Tomcat 8.0。

2 系统开发技术

2.1 Spring

随着 Internet 的迅速发展，开发基于 Bowser/Server 的服务端的程序已经成为 Web 服务应用程序开发的重点。而传统的以 EJB为核心的 J2EE 架构对于大多数中小型企业应用来讲，显得过于复杂，而且其复杂的结构在开发过程中会降低开发效率， 增加开发难度。因此一个具有低侵入性，能够让具体的 J2EE 平台技术处于“松耦合”状态的轻量级的 Web 框架是迫切需要。Spring 框架的产生正式是满足了这种需要（陈鑫，2007）。

Spring是一个开源的Java EE应用程序框架，是针对Bean的生命周期进行管理的轻量级容器。该框架的主要优势是分层架构，允许选择使用某个组件，同时为Java EE 应用程序开发提供集成的框架（Bruce Eckel，2003）。

如图1所示。Spring由七个模块组成，这些模块都是比较松耦合的，一些模块的使用根本不需要另一些模块。

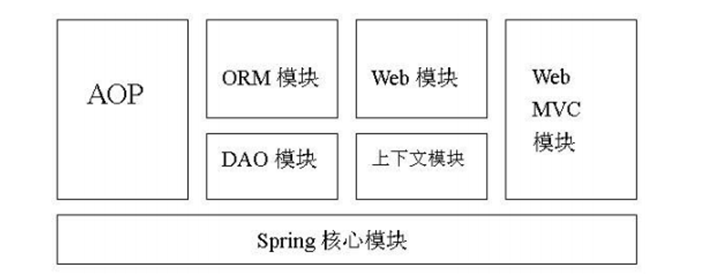


图1 Spring模块划分图

2.1.1 控制反转

Spring的核心概念是控制反转（Inversion of Control），IoC。IoC的Control是控制的意思，其实也是一种依赖关系的转移。意思是高层类不依赖底层。IoC把传统上由程序代码直接操控的对象的调用权交给容器，通过容器来实现对象组件的装配和管理。所谓的“控制反转”概念就是对组件对象控制权的转移，从程序代码本身转移到了外部容器。

2.1.2 面向切面

面向切面编程(Aspect Oriented Programming)，AOP。AOP主要目的是针对业务处理过程中的切面进行提取，所面对的是处理过程中的某个步骤或阶段，以获得逻辑过程中各部分之间低耦合性的隔离效果。可以通过预编译方式和运行时动态代理，实现在不修改源代码的请你工况下，给程序动态添加功能的一种技术。

2.2 Spring MVC框架

Spring MVC使用的是MVC的分层结构（Craig Walls，2011）。MVC模式会导致分离的应用程序（输入逻辑，业务逻辑和UI逻辑）的不同方面，同时提供这些元素之间的松耦合。

（1）模型（Model）主要是根据视图发送过来的数据处理业务逻辑，然后返回处理结果。是完成任务的部分，是整个模型的核心。 封装了应用程序的数据和一般他们会组成的POJO。

（2）视图（View）是模型的外在表现，是和用户交互的界面，是应用系统和外界的接口，一方面接收用户输入的信息，一方面返回模型层处理的数据信息。是负责呈现模型数据和一般它生成的HTML输出，客户端的浏览器能够解释。

（3）控制器（Controller）是模型和视图的联系纽带。接收用户发过来的请求，然后和模型，视图进行匹配，调用模型的方法，然后再根据返回的结果条用视图的页面进行渲染并显示给用户。负责处理用户的请求，并建立适当的模型，并把它传递给视图渲染。

Spring框架（Rod Johnson，2004）提供了构造Web应用程序的全能MVC模块。Spring MVC分离了控制器、模型对象、分派器以及处理程序对象的角色。Spring MVC实现了 MVC的核心概念，它为控制器和处理程序提供了大量与此模式相关的功能。并且当向MVC添加反转控制时，它使应用程序高度解耦，提供了通过简单的配置更改即可动态更改组件的灵活性（薛峰，2012）。

2.3 MyBatis

MyBatis的前身叫iBatis，本是apache的一个开源项目，2010年这个项目由apache software foundation 迁移到了google code，并且改名为MyBatis。MyBatis是一个支持普通SQL查询，存储过程和高级映射的优秀持久层框架。MyBatis消除了几乎所有的JDBC代码和参数的手工设置以及对结果集的检索封装。MyBatis可以使用简单的XML或注解用于配置和原始映射，将接口和Java的POJO（Plain Old Java Objects，普通的Java对象）映射成数据库中的记录。

2.4 MySQL

MySQL是一个关系型数据库管理系统，由瑞典MySQL AB 公司开发，目前属于 Oracle 旗下公司。MySQL 最流行的关系型数据库管理系统，在 WEB 应用方面MySQL是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。MySQL具有功能强、使用简便、管理方便、运行速度快、安全可靠性强等特点（兰旭辉，2004）。

2.5 Android系统

Android是一种基于Linux的自由及开放源代码的操作系统，主要使用于移动设备，如智能手机和平板电脑，由Google公司和开放手机联盟领导及开发。

Android的系统架构和其操作系统一样，采用了分层的架构（余志龙，2009）分为四个层，从高层到低层分别是应用程序层、应用程序框架层、系统运行库层和Linux内核层（墨菲，2010）。

3 系统需求分析

3.1 涉众及系统用例

涉众是某些具有行为的事物，可以使人、计算机系统或组织。表1描述了与本系统应用相关的涉众，主要是用户。

用例图是指由参与者、用例以及它们之间的关系构成的用于描述系统功能的静态视图。系统用例图如图2所示。呈现了用户和系统功能之间的关系。

表1 涉众统计表

|  |  |
| --- | --- |
| 涉众 | 描述 |
| 用户 | 登录，注册，更新用户信息，忘记密码 |
| 宠物犬疾病诊断  疾病诊断反馈 |
| 疾病文章列表查看，疾病文章详细内容浏览  疾病文章推送接收 |

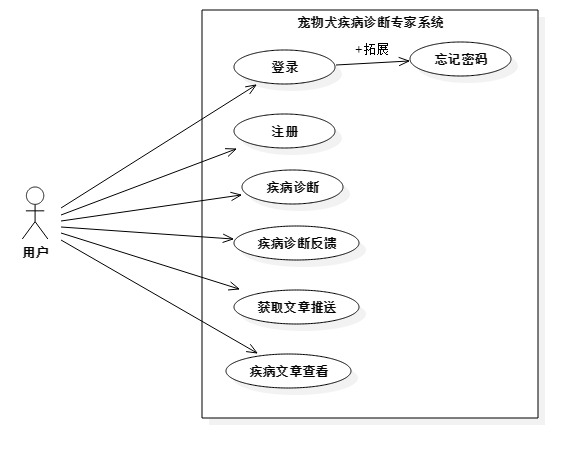


图2 系统的用例图

3.2 系统主要用例

用例是描述了实际情况中相关的各个参与者（涉众）是如何使用系统来实现其各自目标的文本文档，通俗地讲，用例就是一组相关的成功和失败场景的集合（Craig Larman，2004）。表2简要地描述了本系统涉及的七个主要用例。用例的说明使用活动图作为编写用例文本的辅助措施。使用活动图有助于使工作流和业务过程可视化，便于描述复杂工作流的业务用例。

表2 用例统计表

| 用例名称 | 描述 |
| --- | --- |
| UC1登录系统 | 用户在登录系统时输入必要的用户信息（用户名，密码等），即可进入系统使用系统主要功能。 |
| UC2忘记密码用例 | 用户忘记密码想重设密码的时候可以点击“忘记密码”，输入注册的邮箱通过验证即可重设密码。 |
| UC3疾病诊断用例 | 用户根据系统提供的问题进行多次选择，选择后系统返回可能患有的疾病信息和解决方案。 |
| UC4疾病诊断反馈用例 | 用户根据对诊断结果进行评分了留言，对疾病和症状的关系进行优化和提升下次诊断的准确度。 |
| UC5获取疾病文章推送用例 | 系统管理员可以推送和疾病有关的文章，系统也有在一定的间隔内根据诊断的次数结果。用户点击推送结果可以查看到文章的具体内容。 |

3.2.1 登陆系统用例

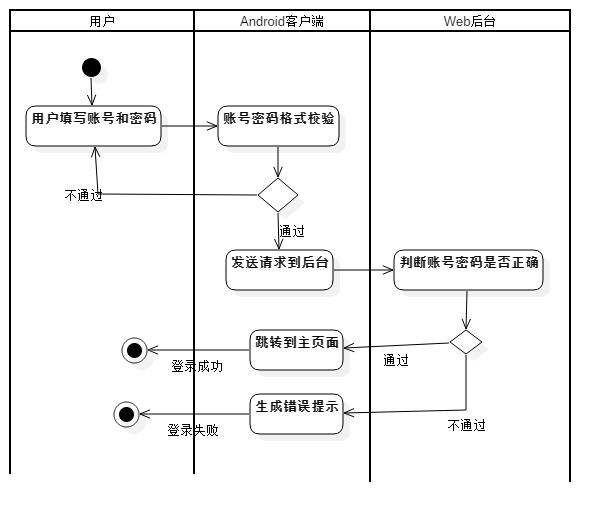


图3 登录系统活动图

登录系统的活动图如图3所示。用例的主要参与者是用户，前置条件是用户已经注册过账号，并且验证通过，成功保证是有正确的账号和密码，系统的主成功场景是用户进入登录界面，系统会要求用户输入登录名或者邮箱和密码。用户输入登录名或邮箱和密码后，系统校验登录信息，判断邮箱格式是否正确，密码位数是否正确。验证通过后系统会发送检验账号密码是否正确的请求到后台，判断成功后即登录成功，系统进入主功能界面。如果用户的账号或者密码出错，则登录失败，系统给出错误提示，要求用户重新输入。

3.2.2 忘记密码用例

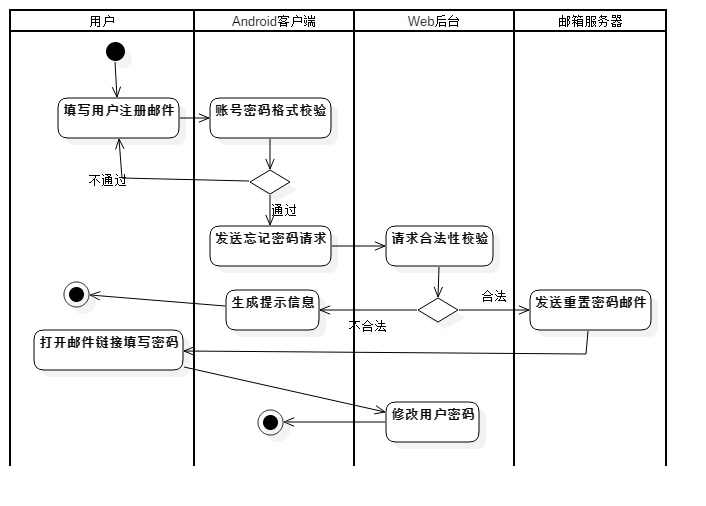


图4 忘记密码活动图

忘记密码的活动图如图4所示。用例的主要参与者是用户，前置条件是用户已经注册过账号，并且验证通过，成功保证是用户已经注册成功并有正确的邮箱地址。主成功场景是用户进入登录界面，选择“忘记密码”功能。系统要求填写注册时候的邮箱地址。用户填写后提交给系统。系统会自动发送重设密码邮件给用户，用户接收邮件并按提示点击所给链接，链接符合系统要求，没有超出有效时间，即可重设密码。

3.2.3 疾病诊断用例

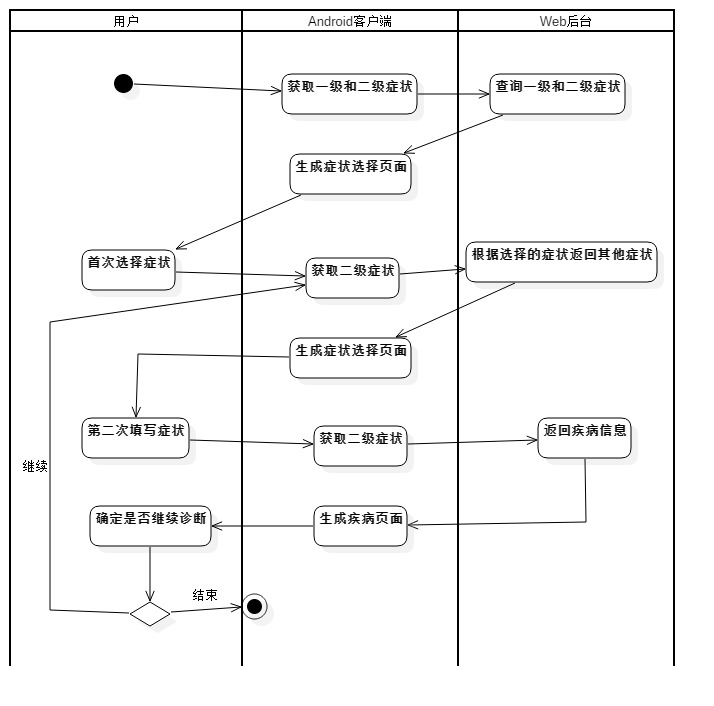


图5 疾病诊断活动图

疾病诊断的活动如图5所示。用例的主要参与者是用户，前置条件是用户已经登录了系统。系统根据一级症状的分类方法给出二级症状给用户进行选择，用户可以根据宠物犬的实际症状进行选择。用户选择完毕并提交给系统后，系统返回多个症状给用户进行二次选择，用户同样根据实际情况进行选择。提交给系统后，系统会根据提交的症状信息返回诊断结果给用户，诊断结果包括几种疾病的名称，简单描述和得相关疾病的概率。用户可以根据情况选择继续提供更多的症状信息。用户重复3-4步，直到选择诊断结束。诊断结束后，用户可以选择给出的疾病，系统会返回相应疾病的详细信息。

3.2.4 疾病诊断反馈用例

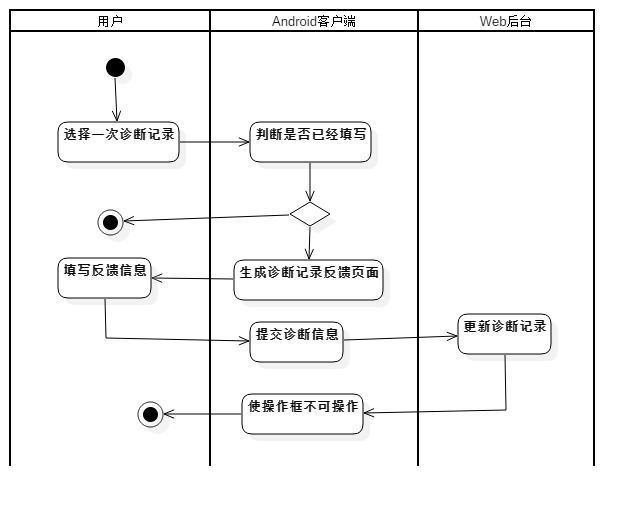


图6 疾病诊断反馈活动图

疾病诊断反馈活动图如图6所示。用例的主要参与者是用户前置条件是用户已经登录了系统，并有未反馈的诊断记录。用户选择诊断反馈功能，进入诊断记录列表。用户选择其中一条诊断记录，系统判断这条诊断记录是否已经填写过，如果未曾填写，那么会进入反馈页面。用户填写反馈信息，包括通过治疗推测的疾病，诊断评分，留言等信息。提交后页面文本框等会变成灰色，用户无法再次提交和选择。

3.2.5 获取疾病文章推送用例

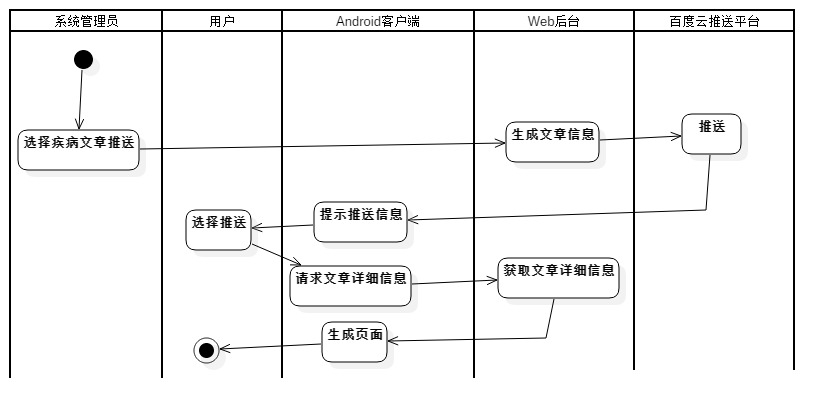


图7 疾病文章推送活动图

疾病文章推送活动图如图7所示。用例的主要参与者是用户，前置条件是用户和管理员已经登录了系统，管理员有推送文章权限。管理员在疾病文章列表中选择其中一篇疾病文章进行推送，推送成功后系统会进行推送结果提示。推送成功后，进入系统的安卓端用户会在手机通知栏收到疾病文章的推送信息，用户点击即可查看内容。

4 系统概要设计

4.1 系统功能和模块的划分

本系统主要分为用户模块、疾病诊断模块、文章推送模块、疾病反馈模块等，表3描述了每个功能模块实现的主要功能。本系统的功能结构如图8所示。

表3 功能模块表

|  |  |
| --- | --- |
| 模块名称 | 功能概述 |
| 用户模块 | 1. 用户注册，登录系统 2. 用户修改个人信息 3. 用户通过邮件找回密码 |

（续上表）

|  |  |
| --- | --- |
| 模块名称 | 功能概述 |
| 疾病诊断模块 | （1） 用户选择症状获取可能发生的疾病  （2） 用户可以查看自己的疾病诊断的历史记录 |
| 文章推送模块 | （1） 系统管理员可以选择疾病的文章进行推送  （2） 用户可以获取到推送信息并查看具体的推送内容 |
| 疾病反馈模块 | 1. 用户可以获取到推送信息并查看具体的推送内容 |
| 1. 系统根据反馈的评分进行规则的优化 |

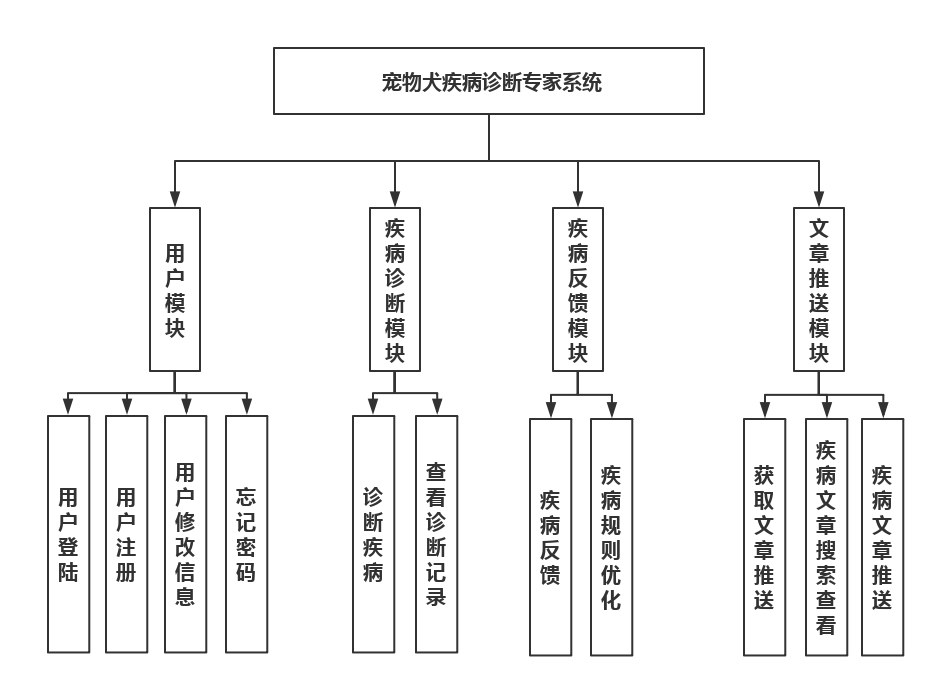


图8 系统模块划分图

4.2 技术架构

4.2.1 Android客户端和Web服务器的通讯

Android和Web后台交互实现主要是用到了http中的get和post请求。Web端在后台管理部分已经有一些接口提供给网页前台，Android客户端同样可以用到。但在一些地方仍然需要编写Android的接口。后台框架用的是Spring MVC，因此接口写在不同模块的controller控制器里，通过访问特定的链接可以返回相应的数据或者提交数据后台进行业务处理。Android端只需要根据格式发送http请求，获取服务器返回的数据即可。

Android端用的是标准Java接口(java.net)，HttpURLConnection，可以实现简单的基于URL请求、响应功能。在程序中，根据请求的特点封装了一个Ajax的类来模拟网页端的ajax进行请求。由于Android4.0后所有网络方面的操作都不能再主线程。Android发送请求需要用到异步处理，因此发送请求的方法用到了回调函数作为参数。异步处理主要用到Android系统的AsyncTask。AsyncTask是Android提供的轻量级的异步类，可以直接继承AsyncTask，在类中实现异步操作，并提供接口反馈当前异步执行的程度(可以通过接口实现UI进度更新)，最后反馈执行的结果给UI主线程。服务器返回的格式默认是用了一个自定义类ResponseData进行封装，然后转成json模式进行发送。客户端接收会把json数据解析成ResponseData，在调用回调函数之前先进行简单的处理，判断后台逻辑是否能正确执行，如果不能正确执行，发生了未知异常，那么客户端会进行统一处理。

4.2.2 图片存储技术

对于头像等图片的存储使用的是七牛云存储。七牛云存储，是专为移动时代开发者打造的数据管理平台，为互联网网站和移动App提供数据的在线托管、传输加速以及图片、音视频等富媒体的云处理服务。

图片上传的时候会把图片进行裁剪，保存在手机中，再上传图片到七牛云服务器上。需要图片的时候只需要图片名，系统通过在手机本地的缓存文件夹进行查找，如果没找到，会构造图片链接通过HttpURLConnection的方式访问七牛云服务器，把图片下载到本地缓存文件夹中。

4.2.3 云推送

Android推送用的是百度云推送平台。百度云推送（Push）是一站式APP信息推送平台，为企业和开发者提供免费的消息推送服务，开发者可以通过云推送向用户精准推送通知和自定义消息以提升用户留存率和活跃度。

在疾病文章推送的时候会用到移动端和服务器建立长连接，使用XMPP长连接，称Push（推送）。长连接是在用户与内容服务商之间建立一种长时间的连接，服务端一旦有新的消息立刻发送给客户端。长连接的长效维持主要通过定时和多循环来实现，Android系统中定时运行过程中有两种不同的方法：①Timer ②AlarmManager。Timer保持唤醒状态的过程中用电量比较大，这在很大程度上降低了终端在运行过程中的待机状态，显然无法满足环境的要求。

AlarmManager主要是对 RTC模块进行实时性管理，它实质上是对时间进行计算的硬件，这种硬件较为独立，PU休眠时它依然能够保持正常的运行状态。由此，当 AlarmManager在进行任务的执行的过程中，CPU能够正常休眠，即可节省电量（刘忠群，2016）。

系统主要通过三方推送平台，百度云推送实现长连接并进行推送。打开Android应用后会启动一个PushService当它在运行时它是和服务器相连，保持长连接。当服务器端有新信息了，则把最新的信息Push到客户端上，通知Notification，PushMessageReceiver接收到信息再进行处理。

4.2.4 邮件服务端

HMailServer是一个运行于微软Windows系统、基于GPL授权、免费的电子邮件系统。HMailServer支持常见的电子邮件协议SMTP、POP3、IMAP，可以很容易地与许多现有的网络邮件系统集成和二次开发。HMailServer内置MySQL/MSSQl数据库，支持多网域，多用户。

系统使用自己搭建的HMailServer邮件服务器进行邮件的发送。使用system@xin.com作为系统通知邮箱，通知用户进行密码重置，系统通知等操作。

4.3 数据库设计

数据库设计遵循第三范式的设计标准，并适当增加冗余字段提高搜索速度。在数据冗余和处理速度之间找到合适的平衡点。在一些需求变动可能相对较大的表中增加了attribute1，attribute2等冗余字段，方便日后数据库变动对代码的修改。

4.3.1 E-R图设计

4.3.1.1 用户模块

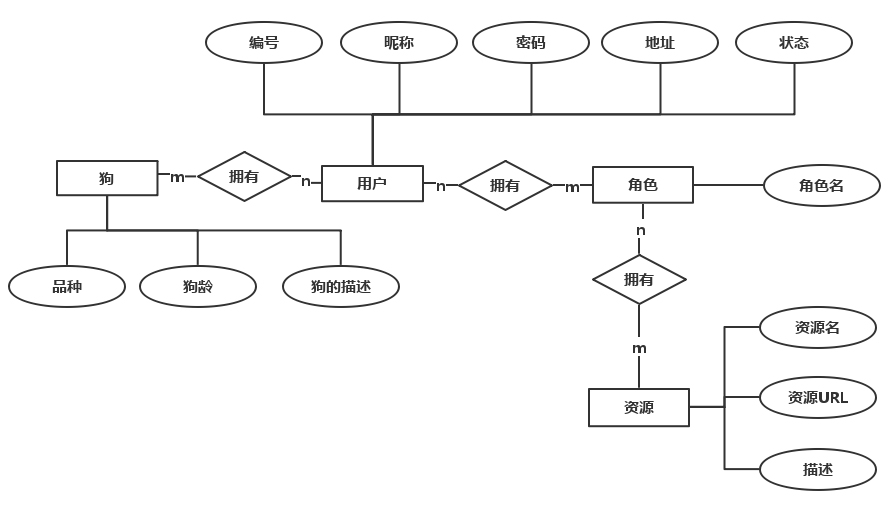


图9 用户模块ER图

用户模块ER图如图9所示。根据需求，用户模块的设计需要增加宠物狗的关联。人和狗是多对多关联，宠物狗的表存储了一般宠物狗的基本信息，关联表存储了用户的宠物犬的基本信息。

宠物狗的关联有助于诊断的时候用户选择并确定狗的信息，提高诊断的准确性和方便后面做宠物狗疾病的分析和统计。

在做登录注册的时候需要到权限控制部分的功能，因此需要和权限表相关联。权限的设计是权限表和资源表的多对多关联，用资源表来记录用户可访问的资源链接。

用户的表设计是分开了user和member表，user表存储的是用户的基本信息，member表存储的是基础用户的拓展信息。这样设计的目的是为了日后可以拓展不同类型的用户，增加兽医等角色。

4.3.1.2 疾病症状模块

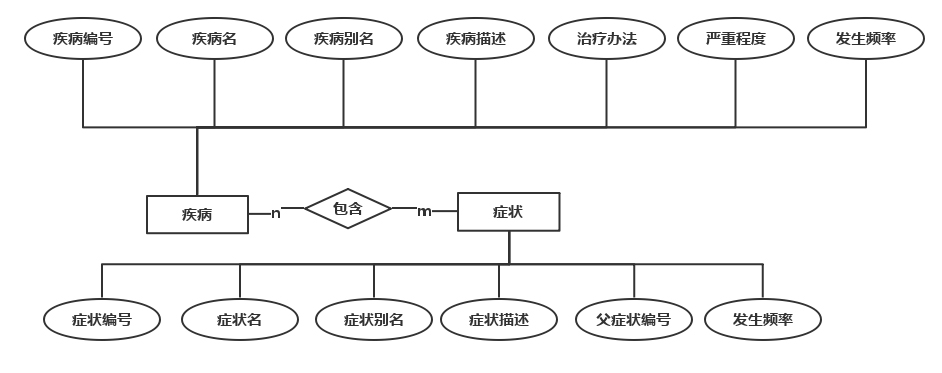


图10 疾病症状模块ER图

疾病症状模块ER图如图10所示。症状和疾病的关系是多对多的关系。针对宠物犬疾病诊断的特点，本研究设计了基于数据库的规则库，在规则库的建设中，用一位数组库来模仿了闹信息的存储和激活方式，以这种方式建立的规则库适合动物疾病诊断知识的特点，具有实用性强、操作和维护简单的优点（王靖飞，2002）。

动物疾病诊断知识的数量是相当大的，有些疾病的症状可能有数十个之多，并且经常出现多个疾病具有共同症状的情况，这给我们设计疾病诊断规则带来了相当大的困难，一是规则数量相当多，因为每种疾病症状的出现随动物的个体状况、疾病的发生时间和许多环境因素的不同而不同，我们要实现对疾病的正确诊断，就需要对各种可能以规则的形式表示出来，更多的情况下症状和疾病的发生之间的关系存在着症状的排列组合问题，所以要表示清楚疾病与症状之间的关系非常困难。另外，疾病之间存在共同症状的情况，使我们在遇到一组症状可以作为多个疾病的诊断条件时无法进行进一步的处理，我们无法确定应该启用那一条规则。根据疾病和症状之间的这种复杂关系和情况，我们对规则进行了一定的扩展，采用如下规则：

IF症状THEN疾病CF

这里的症状是一个单一的症状，CF表示如果宠物犬出现这种症状，对应疾病发生概率。

用这种方法我们就可以表示疾病诊断知识的各种形式。如下例:

IF呼吸困难THEN肺炎0.4

IF呕吐THEN食物中毒0.3

IF心悸亢进THEN肺丝虫病0.1

IF可视粘膜苍白THEN犬细小病毒病0.7

疾病实体中严重程度属性是所患疾病严重程度的分级。一共分成10级，不同级数会推荐不同的治疗方式，对于不严重的疾病，会推荐用户自己采用简单的治疗手段进行处理，如果值比较大，比较严重的疾病，用户无法自己解决，会推荐用户带着宠物犬去宠物医院进行治疗。

疾病实体中发生频率属性是用户选择症状的次数，用户每次诊断的时候次数都会记录在这个字段里面。当用户需要诊断，根据一级症状的分类返回二级症状的时候，会根据这个字段进行二级症状的排序筛选，多数用户选择到的症状将会优先返回给用户来进行选择，从而提高用户选择的概率。

4.3.1.3 疾病反馈模块

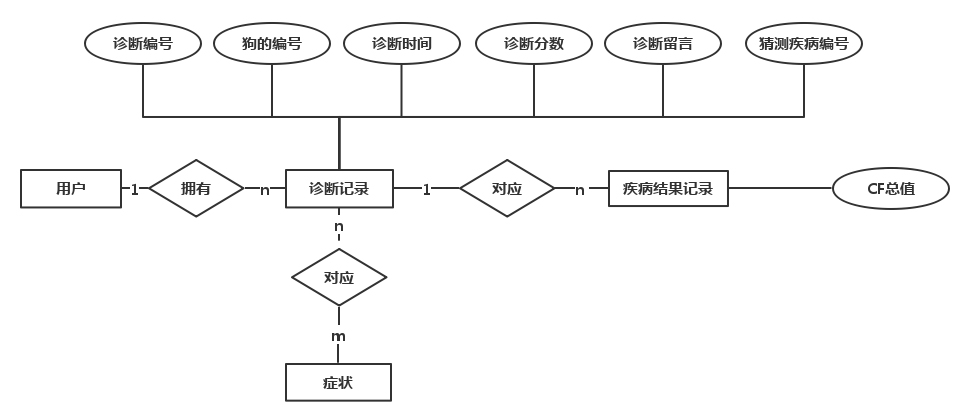


图11 疾病反馈模块ER图

疾病反馈模块ER图如图11所示。疾病反馈表是用于记录诊断的记录，并在记录里面保存用户对诊断过程的反馈信息。主要分成三个表，一个总的诊断记录表，一个疾病记录表，一个症状记录表。诊断记录表和疾病记录表是一对多的关系，和症状记录表也是一对多的关系。当诊断结束的时候会在总记录表插入一条记录，取出改条的id，根据诊断的结果和诊断的选择分别插入疾病记录表和症状记录表中。

4.3.1.4 疾病文章模块

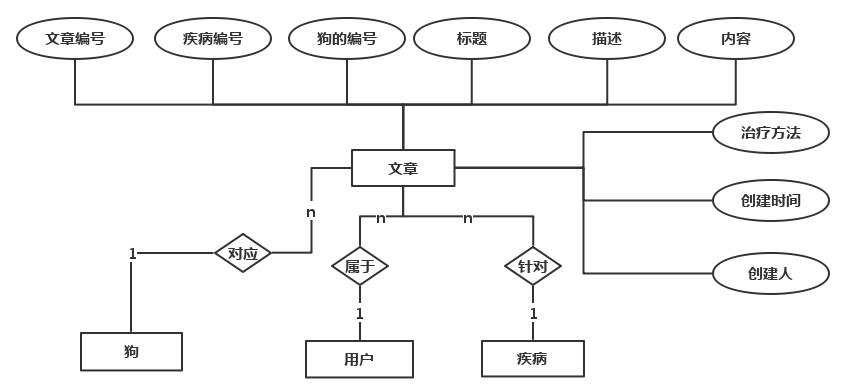


图12 疾病文章模块ER图

疾病文章模块ER图如图12所示。疾病文章的数据库表设计主要是和疾病表和用户表相关联。文章表因为主要围绕着一些疾病进行介绍的，需要到疾病表的一些描述信息和解决方法，因此关联了疾病表，和疾病表一对一的关联。文章表还与用户表相关联，用于判断是哪个系统用户推送的信息。

4.3.2 数据库表设计

数据库表的关系如图13所示。根据实体关系的分析，总共设计出13个表。

表的前缀根据模块划分，分为user、sys、es，分别对应用户模块，系统模块，专家系统模块。

（1）系统模块的功能主要是后台的管理和配置。包括sys\_resource是资源表，存储了系统中的请求链接（resource），父资源id（pid），资源状态（status），资源描述（description）。sys\_menu是后台菜单表。

（2）用户模块中的表有user\_role，user\_user，user\_role，user\_resource。user\_user是用户基础表，储存了用户的基本信息，如账号、邮箱、密码等。user\_role是角色表，储存了不同角色的名称，和sys\_resource表结合起来用作权限控制。

（3）专家系统模块的表有es\_article\_push，es\_dog，es\_disease，es\_symtpom，es\_disease\_symptom，es\_judge\_symptom\_log，es\_judge\_log，es\_judge\_disease\_log。

es\_article\_push是疾病文章表，保存了疾病文章的信息如文章标题（title），文章描述（description），文章内容（content）等。title和description字段是Android推送的时候在安卓客户端通知列表上显示的信息。content字段为推送的自定义内容，由管理员进行编写。

es\_disease，es\_symtpom，es\_disease\_symptom分别对应疾病表，症状表，疾病症状表。疾病症状表保存的是症状疾病的规则信息，这三个表主要用在用户疾病的诊断功能。在表es\_symtpom中发生频率（frequency）字段的作用是用户选择症状的次数，用户每次诊断的时候次数都会记录在这个字段里面。当用户需要诊断，根据一级症状的分类返回二级症状的时候，会根据这个字段进行二级症状的排序筛选，多数用户选择到的症状将会优先返回给用户来进行选择，从而提高用户选择的概率。

es\_judge\_symptom\_log，es\_judge\_log，es\_judge\_disease\_log分别是症状记录表，诊断记录表，疾病记录表，三个表主要用在诊断反馈功能。

es\_dog是保存宠物犬的信息。宠物犬和用户是多对多的关心，es\_user\_dog为宠物犬表和用户表的关联表。

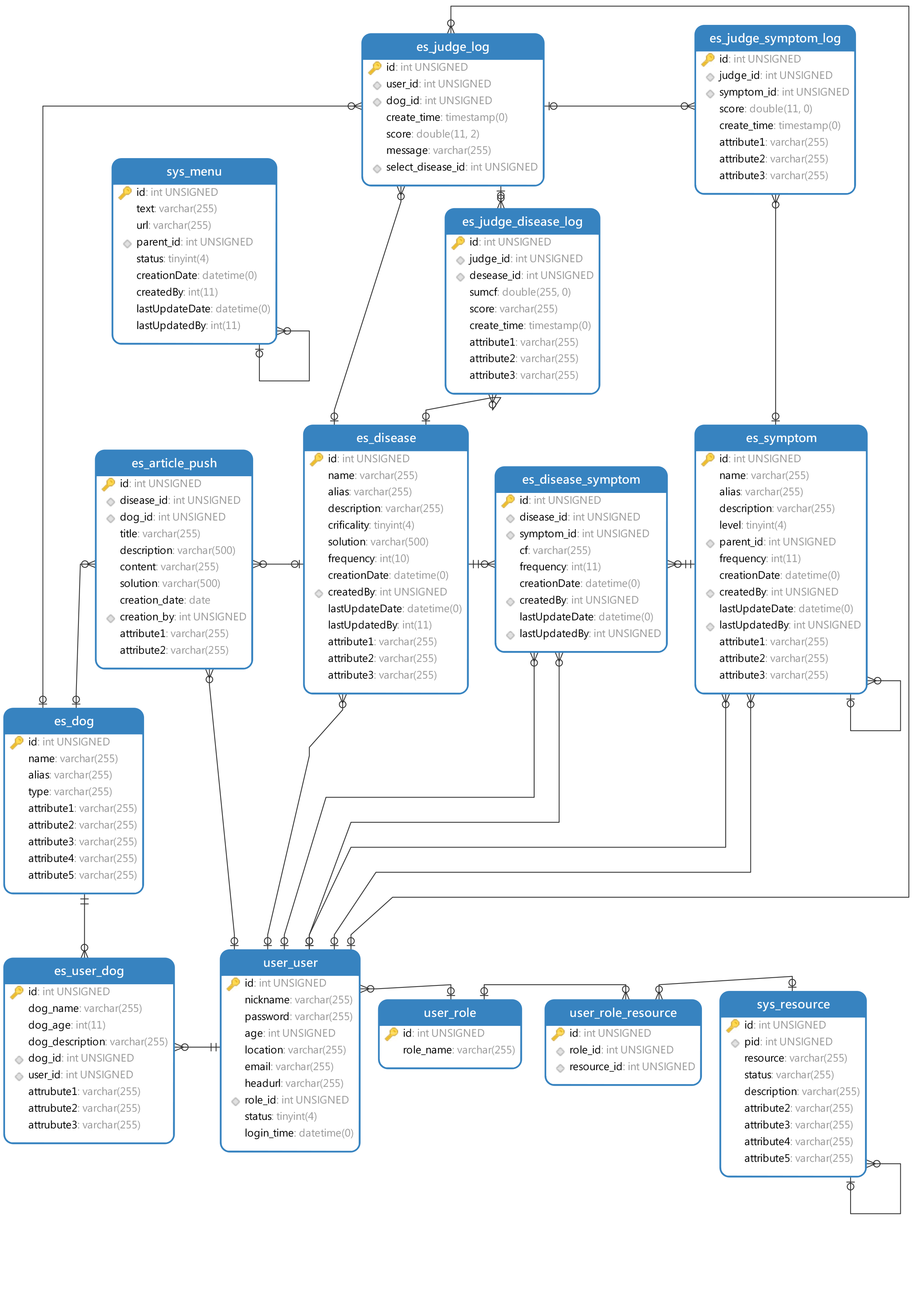


图13 数据库表的关系

5 系统详细设计

5.1 注册登录模块

登录注册是系统的基础模块，用户通过注册并且登录的方式来获取访问系统的大部分功能的权限。

5.1.1 用户登录

已经成功注册的用户，在进入系统之前，需要进行登录操作。用户登录顺序图如图14所示，类图如图15所示。在打开Android应用程序后，会出现登录的页面。用户在页面上输入了用户名（nickname）或者邮箱（email）和密码（password），确定之后，系统在会在前端作一些简单的校验，验证输入的字段是否完整以及符合正确的字段格式，如果不通过，系统会提示错误信息，并要求用户重新输入。检验通过后LoginActivty会封装成User对象，并调用UserService里面的login方法，UserService接着会调用Ajax类的方法进行http请求。

Ajax是一个使用异步请求发送http请求的类，内部使用HttpURLConnection进行请求。SendPost方法能把各种类型的对象封装成参数对象，然后转换成String类型的post参数，封装http请求，并且把请求结果封装成ResponseData对象进行返回。

Service通过回调函数的方式去处理请求完毕的结果，当请求完毕时，把ResponseData对象转换成成User类型的对象，作为参数调用回调函数。

客户端在接收到结果后，进行一些请求结果判断，判断返回的数据是否是正确的，如果是正确的，则会返回给UserService，UserService会调用回调方法，也就是进入主功能Activity中。

登录成功后，后台会设置session来保存当前登录的用户信息，并返回session的id给Android端。因此处理返回登录的返回信息如果有返回sessionId，则会保存在静态变量中。之后每次提交请求都会附带sessionId，用于后台记录当前会话。当sessionId失效时，也就是登录用户超时，返回的Content-Type会为text/html，则跳转到LoginActivity让用户重新登录。

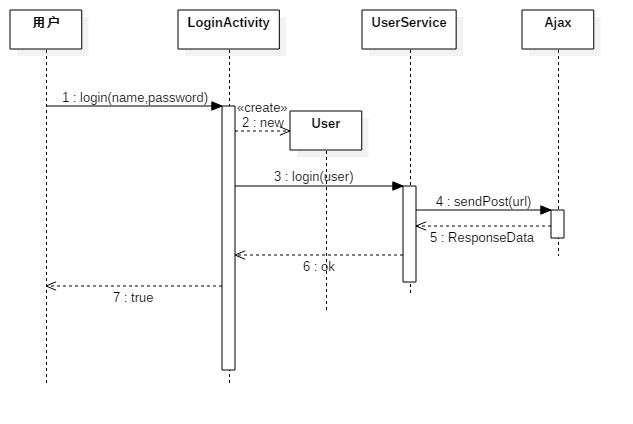


图14 用户登录顺序图

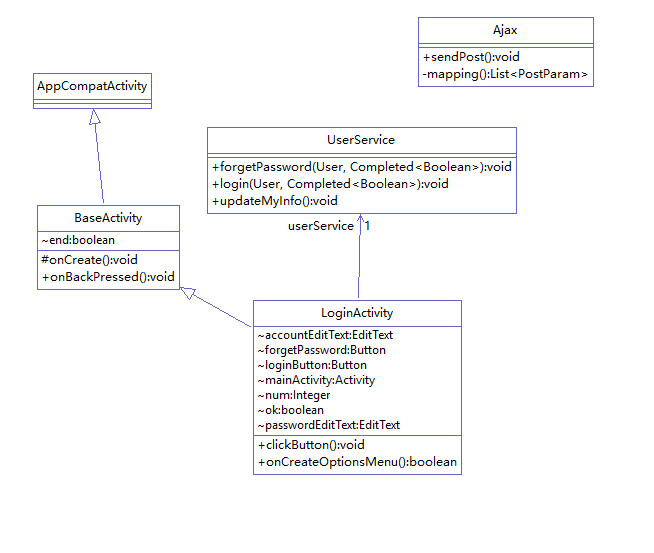


图15 用户登录类图

5.1.2 用户注册和信息修改

用户的信息包括头像，名字，邮箱，密码等，因此界面用了几个可编辑的文本框作为输入。对于用户注册和用户信息修改的差别在于提交的对象当中是否包含了用户的id。如果没有id，后台则默认是新增操作，否则是更新操作。

用户注册和信息修改顺序图如图16所示，类图如图17所示。系统根据不同的组件，获取到用户填写的信息后会把用户信息封装成User对象，再调用UserService的updateMyInfo，updateMyInfo方法里面会首先对名字，邮箱，密码等信息做基本的格式校验。如果不通过，则调用相关组件的信息提示方法进行提示。否则会调用Ajax类中的sendPost方法，把信息提交到后台。后台进行处理后会返回json串，系统把json串转换成，ResponseData对象，判断是否成功。如果ResponseData对象中isSuccess为true，那么系统提示更新成功，否则提示在ResponseData对象中error属性，error属性带有错误提示信息。

对于头像等图片的上传存储使用的是七牛云存储。在修改用户信息的界面上，按头像的会调用startActivityForResult跳转到android.provider.MediaStore.Images.Media. EXTERNAL\_CONTENT\_URI功能，调用系统的相册功能，打开相册，并且相册中有拍照功能，让用户可以选择图片，或者选择重新拍照。拍照或者在图库中选择图片后系统会调用startActivityForResult跳转到com.android.camera.action.CROP，进入图片裁剪的页面。选择确定后，后台会取出系统缓存的裁剪后的图片内容BitMap，调用BitMap中compress方法生成jpg文件，保存在sd卡中，并且会上传到七牛云服务器中。上传成功后会调用回调函数，把图片显示在页面上。当确定修改信息后，会同时更新左侧主菜单几面中的头像显示。

当需要获取头像的时候，可以根据头像的名称，调用QiNiuUpLoad中getPicture的方法，该方法会首先从本地sd卡查看是否有同名的文件，如果有则调用BitmapFactory.decodeFile(file.getAbsolutePath())，把文件转换成Bitmap的对象返回，如果没有，则会调用getImageBitmapFromQiNiu方法从七牛服务器上使用HttpURLConnection的方式去下载图片，下载成功后把图片保存在本地sd卡中，并读取文件，返回Bitmap对象。

Android端上传文件到七牛云存储还需要一个秘钥Token。由于Token是根据七牛账号上的秘钥AccessKey，SecretKey，存储空间名和Token的有效时间动态生成的。以你次Web后台中需要增加了一个获取Token的接口。作者根据七牛云存储提供的API把获取Token的方法写成一个jar包，使用了maven进行jar包管理，使用mvn install命令发布到本地仓库中，并把依赖添加到当前项目。Web后台依赖该文件后在Spring中配置accessKey和secretKey等必要信息，在控制层增加QiNiuUpLoadController的控制器，即能提供获取Token的接口。

Andoird端每次上传文件前，都会访问接口从服务端SDK获取Token，然后调用七牛云存储的Android端API接口进行图片的上传。上传完毕后会设置一个变量作为上传文件的名字，默认会把用户id加上时间戳作为文件名以保持图片名字的唯一性。用户更新信息的时候会把这个变量作为头像地址一起提交到Web后台服务器更新。下载的时候只需要提供头像名称，即可把文件从本地进行查找，如果本地没有，则从七牛云存储服务器中下载下来，保存在本地。

对于其他模块一些图片的上传下载如疾病文章的图片显示，同样会用到上面的方法进行统一的处理。

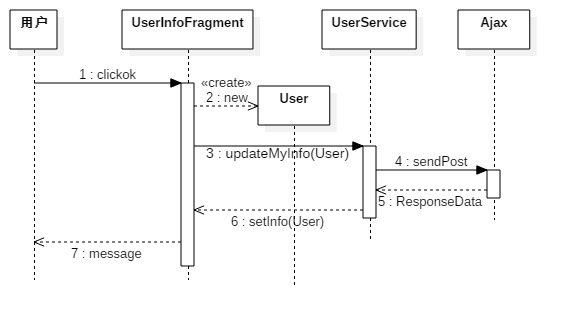


图16 用户注册和信息修改顺序图

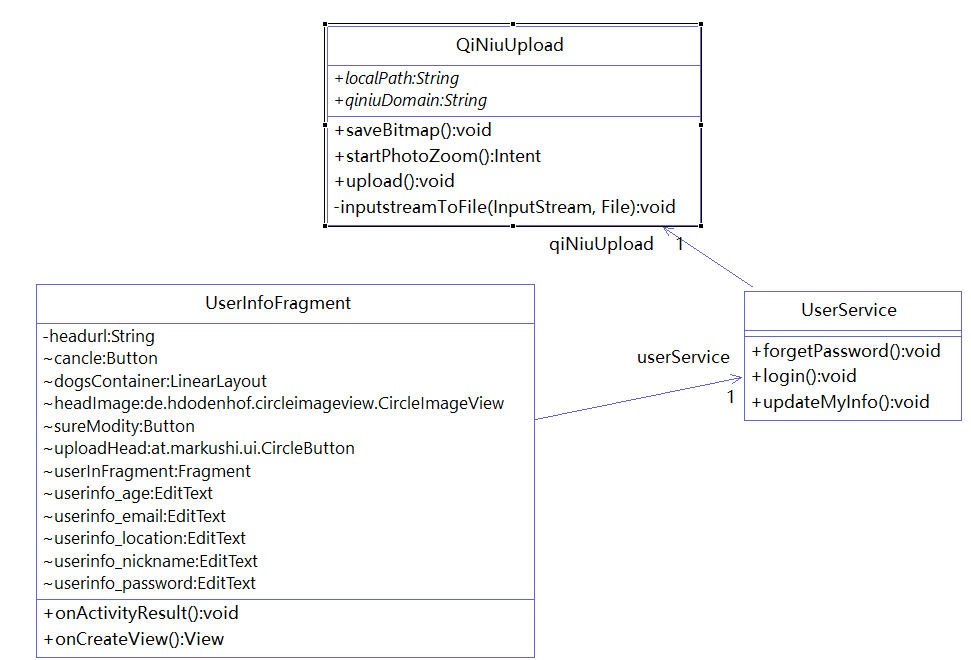


图17 用户注册和信息修改类图

5.1.3 忘记密码

忘记密码的顺序图如图18所示，类图如图19所示。用户在登录界面可以选择“忘记密码”操作。点击后会出现邮件输入框，用户输入后点击确定。系统首先会对邮件的格式进行校验，如果校验成功，则会调用UserService的forgetPassword方法，把用户输入的邮箱信息发送到Web后台中去，并用AlertDialog构造提示框，提示用户进入邮箱查看修改密码邮件。用户点击提示框后，界面跳转LoginActivity，重新进入登录界面。登录界面的账号会自动变成用户在忘记密码时填写的邮件。

邮件服务器用的是在自己本地上搭建的hMailServer，system@xin.com作为系统发送的邮箱。在Web端用的是javax.mail进行邮件的发送，作者把邮件发送的方法写成jar包安装到本地maven仓库，添加依赖到本项目，发送邮件时只需要调用Email类中sendEmail，把用户邮箱和发送内容作为参数即可。重置的形式为点击邮件内的链接，进入修改密码的界面，修改后会发送链接“user/resetPassword”到后台。邮件中重置密码链接格式为petdogexpert/user/resetPasswordShow?key=xxx@xin.com!!1460265599024用户邮箱加时间戳作为参数。时间戳是发送邮件的时间，由于暂时未把key加密，因此以明文的形式显示。当用户填入“密码”和“再次输入”后，会在前端作一个校验，判断这两个文本是否相等，如果相等，会把密码和链接中的key值作为post参数发送ajax请求到后台。

后台接收到会调用UserController中的resetPassword，resetPassword会执行UserService中resetPassword方法。UserService中resetPassword方法主要是对key值进行校验，通过对key值的分割处理，获取到email和时间戳，先判断当前时间戳key时间戳是否小于3600000，也就是时间间隔是否在1小时内，如果在一小时之内打开链接，则是在有效期内，可以调用UserMapper中updatePasswordByEmail，根据email查找出数据库的一条数据进行并对password字段进行更新操作。如果超出有效期，则链接无效会抛出异常，在前端页面中提示邮件已经过了有效期，无法修改密码。

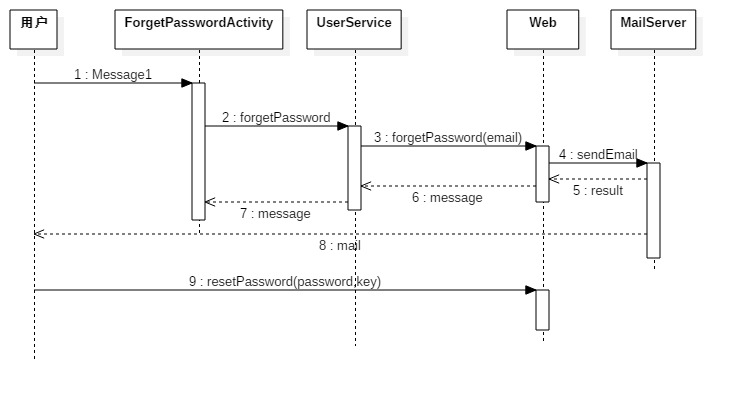


图18 忘记密码顺序图

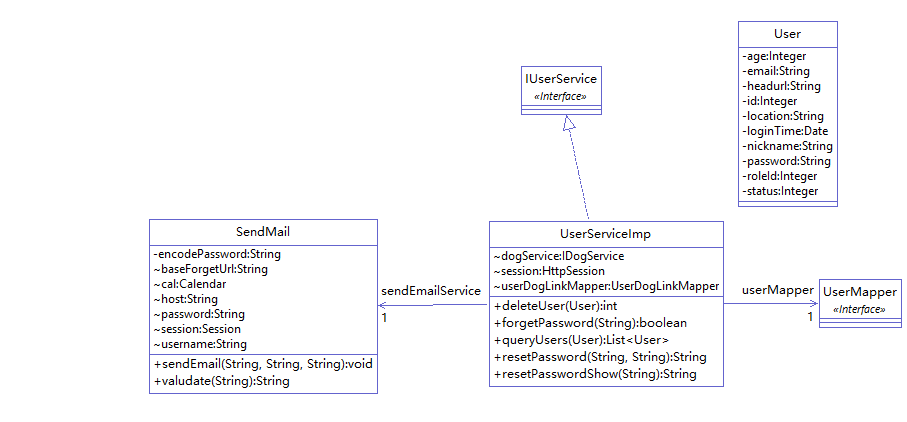


图19 忘记密码类图

5.2 疾病诊断模块

疾病诊断是系统主要的功能模块，负责通过和用户交互的方式获取宠物犬的症状，然后返回疾病信息。

5.2.1 疾病诊断流程

疾病诊断顺序如图20所示，疾病诊断类图如图21所示。用户进去疾病诊断功能后，系统会调用ExpertService里面的getFirstSymptoms，这个方法的作用是获取第一次显示的症状信息。第一次显示的症状分为一级症状和二级症状，一级症状起到分类的功能，二级症状起到给用户选择的功能。getFirstSymptoms方法会调用Ajax类的静态方法sendPost，发送请求给Web后台进行处理。后台会返回症状信息给ExpertService，ExperService会根据获取到的结果渲染页面，然后显示给用户。

用户进行完第一次选择后进行提交，提交后系统会获取用户在页面上选择的症状，并封装成List<Symptoms>，作为参数调用ExpertService里面的getFirstCheckRet (symptoms)方法。GetFirstCheckRet方法同样会调用通用的Ajax类的sendPost静态方法去访问后台，获取提供第二次选择的症状信息返回给ExpertService，ExpertService会根据症状信息进行页面处理，让用户进行二次选择。

用户进行完第二次选择后提交，提交后系统会再次获取用户在页面上选择的症状，调用ExpertService里面的getNextCheckRet方法，这个方法的作用是根据前两次用户选择的症状来获取从Web服务器返回的疾病信息。ExpertService获取到疾病信息后会返回到页面中去。用户可以查看到诊断结果，并且可以进行下一步的选择来使得诊断结果更精确。

进行多次选择其实也就是调用了ExpertService的GetFirstCheckRet方法，向服务器获取更多其他的症状信息供用户选择。

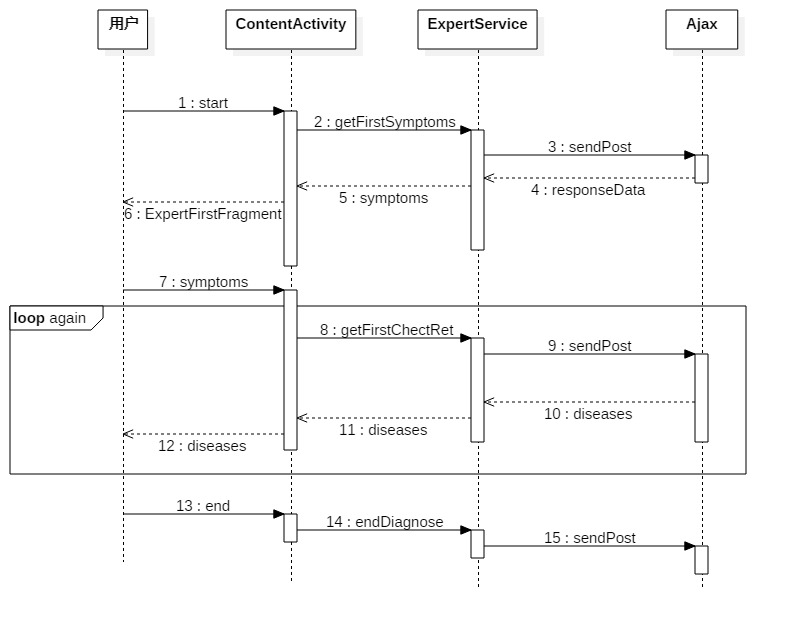


图20 疾病诊断顺序图

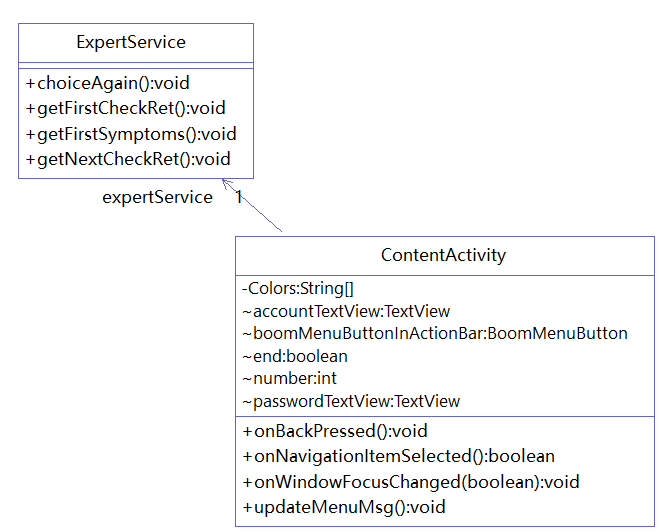


图21 疾病诊断类图

前端主要是发送三个不同请求给后台来返回数据

1. getFirstSymptoms对应的请求是“/expert/getFirstSymptoms” SymptomService会调用getLastTreeSymptoms获取所有一级症状和一级症状下面按frequency排序前10的二级症状。
2. GetFirstCheckRet对应的请求是“/expert/getFirstCheckRet”前端的参数包括了用户选择的症状和用户没有选择的症状。ExpertService获取了参数之后会调用addSymptomFrequency方法把选择的症状对应的frequency进行+1操作。然后根据用户选择的症状在数据库中获取相关联的疾病信息，并根据选择的症状统计疾病的CF总值。对疾病进行CF值从大到小的排序，获取前5个疾病。然后在数据库中获取和这5个疾病相关联的所有症状，除去重复的症状，并按CF值的大小进行排序，返回小于10条的症状信息返回给用户。对于已经选择的症状后台会保存在session里面，不会重复返回相同的症状给用户。
3. getNextCheckRet对应的请求是“/expert/getNextCheckRet”ExpertService会调用getNextCheckRet方法，主要是根据之前的症状选择返回诊断结果。系统会根据之前选择的症状在数据库中查找与症状相关联的疾病，并把关联的症状的CF值相加。并且把统计好的疾病信息进行排序处理。最后返回给用户。
4. 用户可以选择继续选择症状，会发送“/expert/choseAgain”请求。Web服务器内部会重新调用getFirstCheckRet方法，系统会返回更多的症状给用户进行选择。

（5）如果用户选择结束诊断，会发送“/expert/finish”请求到Web后台ExpertController接收到请求调用finish方法，finish方法会调用ExpertService中的 finish方法，service方法主要是对诊断过程进行记录，在es\_judge\_log中插入一条数据作作为诊断记录，从session里面取出用户选择的症状和得出的疾病信息，写入数据库的es\_judge\_disease\_log，es\_judge\_symptom\_log表中。最后清除session中的诊断记录。

5.2.2 疾病诊断反馈

疾病诊断评分是在完成了一次诊断后对诊断结果进行的反馈以优化症状和疾病的规则。用户在菜单中点击诊断反馈即可查看自己的所有诊断记录，以列表的形式呈现出诊断记录的基本信息。

列表的组件同样用了开源的pulltorefresh组件，可进行下拉更新和上拉加载的操作。选择其中一次的诊断记录即可进入诊断评分的页面。

对于诊断反馈主要的操作是:

（1）在诊断结果的疾病建议中，根据用户的实际治疗情况进行判断，判断最有可能发生的疾病进行选择。疾病的呈现形式是用一个饼图，饼图是用了开源框架MPAndroidChart里面的PieChart，并根据API对PieChart作定制化的修改显示。

（2）对诊断结果进行评分，用的是Android的原生组件RatingBar。

（3）对诊断进行留言评价。

当用户对上面三个反馈进行操作之后，确定后即会把评价信息封装成JudgeLog对象，有Ajax的sendPost方法发送请求给Web后台。后台接收到请求后根据收到的信息在数据库表中进行更新操作。

更新成功后，系统会调用setViewsDisabled方法，该方法会将PieChart，RatingBar，留言评分中的EditText设置为disabled，不可操作，禁止用户再进行修改提交。

5.3 疾病文章推送模块

5.3.1 疾病文章推送流程

Android推送用的是百度云推送平台。在打开应用程序的时候会建立一个名为Push Service的服务，这个服务会自动和百度云推送平台建立一个长连接。

疾病文章推送顺序图如图22所示。当进行推送的时候系统会发送推送信息给百度云推送平台，推送平台会发送推送信息给Android客户端，客户端会以接收广播的方式调用继承了百度云包内BroadcastReceiver的PushMessageReceiver的方法进行接收，接收后会在手机的消息通知列表显示推送信息。

用户点击后会调用接收器的onNotificationClicked 方法，方法会获取广播信息中疾病文章的title，description和id等少量信息，然后封装成对象并跳转到首页，首页根据获取到的信息会再次跳转到ArticleActivity，ArticleActivity会接收到id，然后调用ArticleService中getArticleDetail方法，getArticleDetail会调用Ajax的sendPost方法去后台根据文章的id获取文章的详细信息，并进行返回，返回结果后getArticleDetail会根据回调函数处理界面，把内容显示在页面上。

后台的推送方法是基于百度云推送的Java服务器端的SDK进行处理的。当系统管理员决定推送一篇文章到Android客户端的时候，会调用后ArticleController控制器中pushToAndroid方法，控制器会获取文章的id并调用ArticleService的pushToAndroid方法，首先会调用getArticleBase方法获取疾病文章的title和description等基础信息，封装成推送所需要的对象，再转换成json字符串，然后调用pushToAndroid方法。pushToAndroid里面的方法的实现是基于百度云云推送的官方用例进行编写，打包成jar文件，通过maven进行管理，添加到本系统中，加上应用的APIKEY和SECRETKEY并调用API即可发送推送请求给服务器，由百度云推送服务器推送到Android客户端中。

当用户需要展示文章具体内容的时候会发送请求给后台，后台会根据获取到的文章id在数据库中获取该文章的详细信息进行返回。疾病文章部分用户还可以进行点赞操作，点赞的冒泡效果使用了heartLayout开源组件。

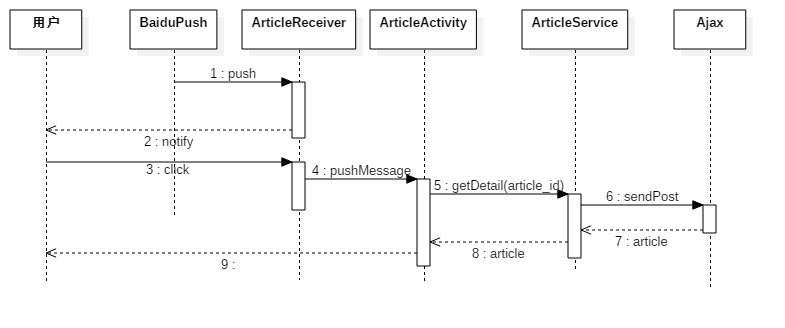


图22 疾病文章推送顺序图

5.3.2 疾病文章列表获取

在做疾病文章列表展示的时候用的是android-pulltorefresh这个开源项目。android-pulltorefresh 是一个强大的拉动刷新开源项目，支持各种控件下拉刷新。

疾病文章列表获取类图如图23所示。在疾病文章列表的功能中主要用到了下拉刷新onPullDownToRefresh和上拉加载onPullUpToRefresh的功能。

onPullDownToRefresh：这方法是用来刷新的，当用户下拉到一定程度的时候，会自动调用这个方法。这个方法会调用ArticleService的getArticles方法发送请求到后台获取第一页的文章信息，当获取成功的时候会调用回调函数，回调函数会清空adapter(ListView适配器)和mArticlePushList（保存基础数据的列表），并把获取到的文章数据插入到mArticlePushList中，重新设置适配器来清空缓存。

onPullUpToRefresh：这方法是用来加载更多数据的，当用户在列表底部继续上啦到一定程度的时候会自动调用这个方法。同样会调用ArticleService的getArticles方法，发送的参数是最旧那条数据的id，后台接收到id会查询比这个id小的行数据，进行分页后会取最新的10条数据进行返回。

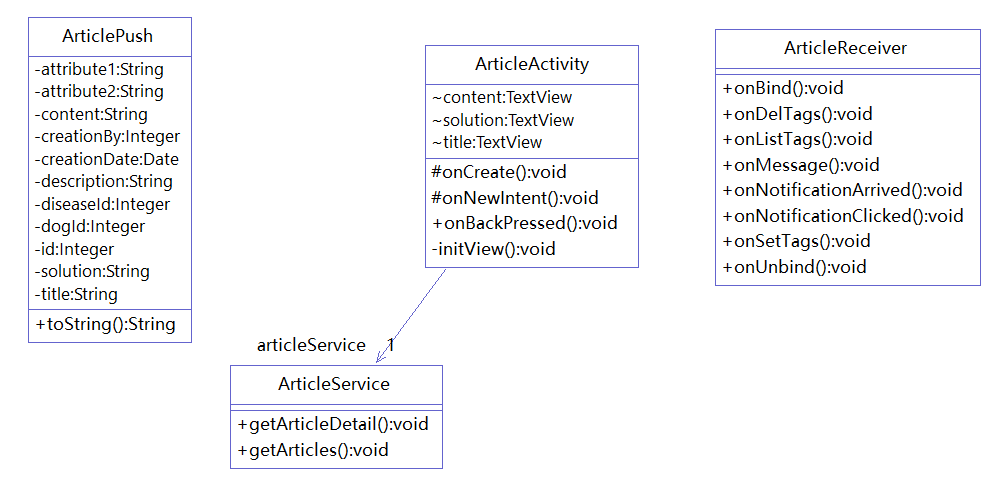


图23 疾病文章列表获取类图

6 系统实现

6.1 用户登录

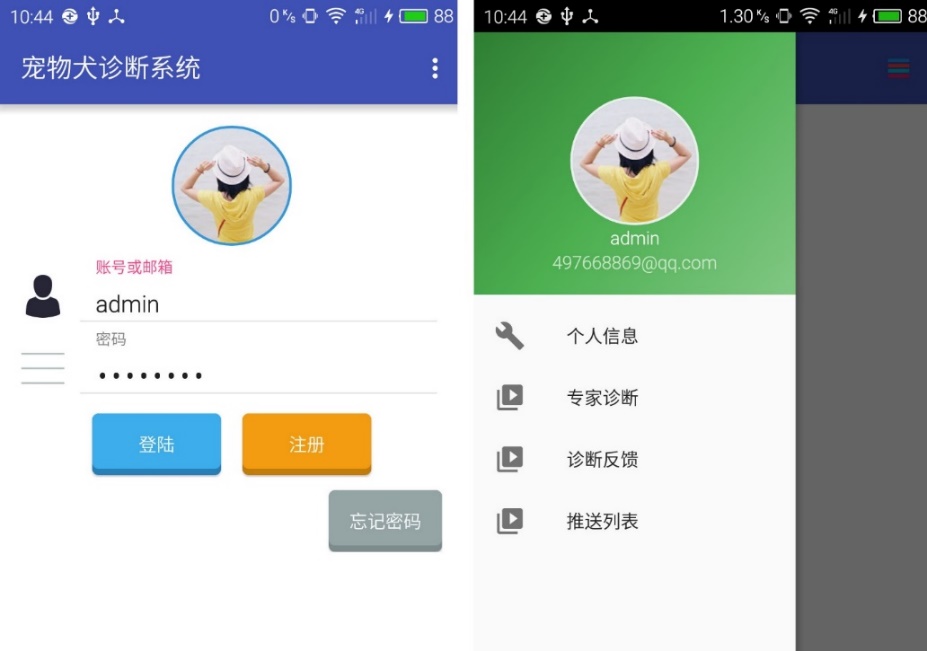


图24 系统登录界面

系统登录实现如图24所示。系统会在本地数据库保存登录过的用户的信息，如果曾经登录系统，输入账号就会自动显示用户头像。输入账号名或者邮箱会有自动完成的功能，显示曾经登录的用户的账号。登录系统后会进入主页面，在主页面左侧进行右拉操作会显示用户基本信息和主菜单。

6.2 疾病诊断流程

疾病诊断实现如图25所示。用户点击一级症状的分类框会弹出二级症状的选项框，填写完毕按确定后，诊断页面下方会显示已经选择的症状。用户可在选择的症状中左拉或者右拉来取消症状的选择。疾病诊断结果如图26所示。



图25 疾病诊断界面



图26 疾病诊断结果

6.2 疾病诊断反馈

疾病诊断反馈实现图如图27所示。页面会显示诊断结果的疾病可能性比例。需要反馈的信息包括，用户经过治疗认为宠物犬会得的疾病，对诊断过程的评分还有诊断留言。填写完毕后确认会使输入框不可编辑。

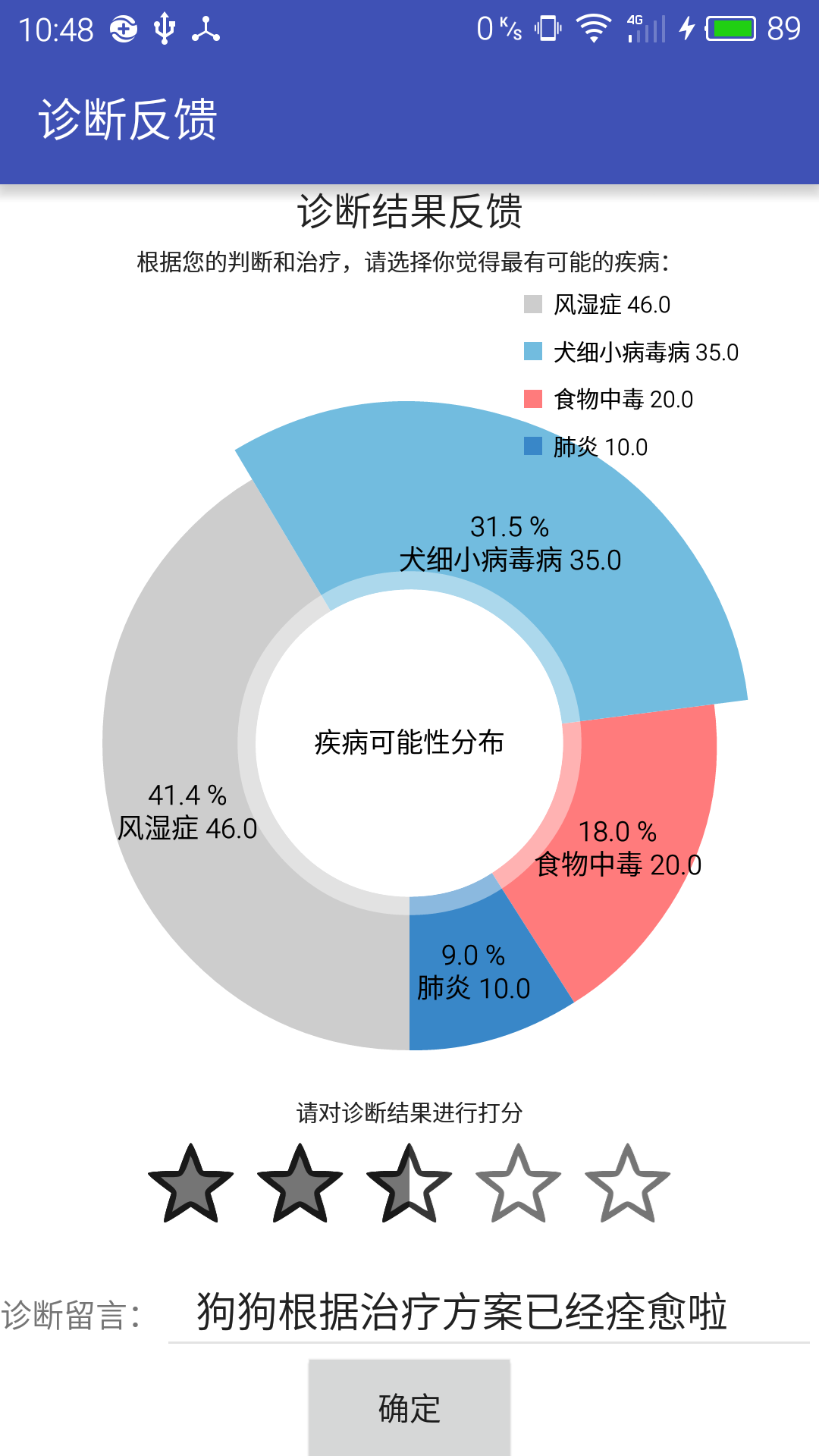


图27 诊断反馈界面

7 结论与讨论

7.1 结论

本文主要对宠物犬疾病诊断流程和专家系统的了解和研究的基础上，实现了逻辑清晰的宠物犬诊断系统。系统采用的是和后台服务器接口对接的形式，保证了Web端和Android端逻辑的一致性，后台主要完成复杂的逻辑，Android端主要用于展示数据，和用户进行交互。

系统的实现过程中运用了面向对象的思想，使用UML实现对业务进行建模，并用工程化的方法进行软件设计和开发。

在本系统中，运用了一些开源的组件进行美化，方便用户进行使用，同时也提高了对用户的友好度。并用Gradle进行项目的管理，实现对开源组件的有效方便的管理。项目用到Git作为版本管理工具，使得多人协同开发变得简单高效。

基于Android客户端的宠物犬疾病诊断专家系统完成了通过选择症状，对宠物犬发生的疾病进行判断。在此基础上增加了用户模块，诊断反馈和文章推送等功能，丰富了系统的内容，也提高系统的可用性，使用户对系统有更好的体验。

7.2 讨论

系统基本实现了宠物犬疾病的诊断，并通过增加文章推送，诊断反馈等模块丰富了系统的功能。但该系统仍然存在缺点和急需改进的地方。

1. 系统的后台管理系统不够完善，功能还不够强大，无法很好的和Android系统的相关功能结合起来。在日后的版本中，会针对后台管理系统进行功能的完善，并提供统计功能统计Android端的一些数据，便于管理者分析。
2. 系统的错误提示并没有完善好。系统发生各种异常的时候没有很好的把它们获取并在界面上给用户进行提示，对于用户来说不知道为什么系统会发生异常的行为，如闪退，按按钮没反应等。在后台方面也没有尽可能细致的把异常抛出并进行处理，这对用户的使用带来了消极的影响。在下一个版本中，后台需要粒度更小的异常处理，Android客户端需要进行更人性化和更完善的的异常提示。
3. 系统存在隐藏的问题，虽然在测试的过程中已经在慢慢把漏洞修复好，但由于测试的次数不够多，且用户量比较少，一些难以发现的漏洞仍然存在于系统当中。系统需要增加漏洞反馈机制，让用户能反馈漏洞，并完善日志的处理流程，及时发现漏洞并且及时修复漏洞。

参 考 文 献

陈鑫. Spring框架的分析研究[J]. 电脑知识与技术:学术交流，2007，3(13)：108-110.

兰旭辉. 基于MySQL的应用程序设计[J]. 计算机工程与设计，2004，25(3)：442-443.

刘忠群. 基于Android平台消息推送机制研究与解决方案的实现[J]. 电脑编程技巧与维护，2016(04)：100-102.

墨菲. Android开发入门教程[M]. 北京：人民邮电出版社，2010.

钱乐秋. 软件工程(第2版)[M]. 北京：清华大学出版社，2013.

王靖飞. 动物疾病诊断专家系统的研究与应用[D]. 沈阳：东北农业大学，2002.

王晓娟. 宠物犬常见疾病防治专家系统的设计与开发[J]. 信息技术与信息化，2014(8)：95-97.

薛峰. 基于Spring MVC框架的Web研究和调查[J]. 合肥工业大学学报，2012(3)：337-340.

小萍. 家犬疾病发生特点与预防[J]. 北京农业，2004，33 (09)：29-30.

余志龙. Google.Android.SDK开发范例大全[M]. 北京：人民邮电出版社，2009.

张煜东. 专家系统发展综述[J]. 计算机工程与应用，2010(19): 55-66.

郑家三. 动物疾病专家诊断系统及其应用[J]. 吉林畜牧兽医，2007，28(5)：17-19.

Bruce Eckel. Thinking In JAVA[M]. Upper Saddle River：Prentice Hall PTR，2003.

Craig Walls. Spring in Action[M]. Greenwich：Manning Publication Co，2011.

Larman. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development，Third Edition，Craig Larman[M]. Upper Saddle River：Prentice Hall，2004.

RE Johnson. Designing Reusable Classes[J]. Journal of Object-Oriented Programming，1988，1(2)：253-286

Rod Johnson，Juergen Hoeller. Expert One-on-One J2EE Development without EJB[M]. Mississippi: Wrox，2004：143-148.

致 谢

毕业设计从无到有，除了自己投入了不少精力去设计开发，也离不开老师同学们对我的帮助。感谢吴春胤老师在我论文完成过程中给我的帮助和关心。他有着严谨的科学态度，渊博的知识积累，强烈的责任心，对我的毕业设计从开始的设计到后面的编码实现都给予了很多合理的建议，对毕业设计的完成和质量的保证有着极大的帮助。

除此之外，我还要感谢我的同学们，在他们的支持和帮助下，毕业设计得以顺利完成，在我遇到生活上，学习上的问题时，他们总是乐意帮我出主意，一直鼓励着我。

最后我要感谢我的家人，他们用最大的付出给予我最大的自由去做我喜欢做的事情，让我能轻装上阵，去追求自己的理想。我所取得的一切都和他们密不可分。