程序设计说明书

基于javaEE+Android+Html5的在图片视频分享系统

# 一 绪 论

**1.1 研究的背景和意义**

互联网技术自从诞生以来已经走过了40多个年头，他是人类发展史上的重大变革，是每个人的免费而又无尽的巨大图书馆，是每个人通向世界的一扇窗，为人类社会带来的便利更是无法用数字来衡量。

2008年以来，智能手机热潮席卷全球。随着移动操作系统的更新迭代，移动应用也像雨后春笋一样：娱乐、音乐视频、生活、社交等等应用应接不暇，渗透到我们生活中的方方面面，给我们的生活，工作和学习带来了无尽的便利。

社交应用，作为移动应用的中坚力量，几乎每个人的手机上都有至少一两款，每个人每天都会不厌其烦的打开一次又一次。各大平台都有很多社交应用，微博，微信，twitter，facebook，instagram等等能更是其中的杰出代表。他的迅速，便捷，廉价等等特点使得这些应用深受人们的喜爱，甚至大有取代主流媒体的驱势。

而且，对于每个人来说，图片视频相对于文字来说更易被人们所接收，图片视频的所见即所得性是单纯的文字永远都无法超越的。

但是，由于各大厂商之间互相竞争，应用在各个平台的移植便成了问题，想要一次开发，更是难上加难。

2008年1月22日。HTML5 第一步草案颁布以来，得到了各大浏览器厂商的支持，各大浏览器已经具备了某些 HTML5 支持。HTML5还在不断完善当中，很多地方还是很不成熟，但是业界一致看好他是未来跨越各平台的桥梁。

在这样的形式下，运用新的HTML5技术开发一款跨平台的用户贡献内容的，以图片和短视频作为传播介质的社交应用，就显得非常有意义。

**1.2 国内外社交应用软件对比分析**

新浪微博：

作为在中国大陆最火的短消息分享应用，拥有庞大的用户群，以简短文字为主题，配有图片视频，实现了阅读、发布、评论、转发、私信、关注等各主要功能，支持本地相机即拍即传和新消息提醒，手机拍照一键发送，支持地理定位，总的来说，功能非常完善。是中国流行最广的社交应用之一，微博不仅仅用于用户联系，很多人运用微博进行寻物启事，打广告等等。

图钉：

基于位置的手机拍照分享社区应用。随手拍下的图片，可以运用滤镜美化，可以添加简短的语音或文字描述；以照片为主题来进行传波信息，可以快速在人与人之间实现信息共享。信息流页面展示关注的用户发布的图片，有分类展示界面，用户可以对选择性的查看自己喜欢的模块，相应模块能对所有用户所发布的信息进行筛选并展示。

Vine：

2013年1月24日，Twitter旗下短视频分享应用Vine上架苹果商店。用户量迅速得到攀升，这都得益于他最大的特点就是短视频拍摄与分享。vine允许用户拍色仅仅6秒的视频，每个用户可以看到别的用户分享的视频，像别的社交应用一样，vine也有信息流页面，有关注功能，没有点击播放概念，滚动屏幕到哪个视频，那个视频就自动播放，像阅读文字和图片一样自然。

**1.3 图片短视频分享应用的价值**

如果没有信息更新，这样的应用毫无生机；如果只是靠管理员来更新信息，这样的

应用信息缺乏及时性，广泛性，准确性；如果通过用户来提供信息，这样的应用才能真正被用户所采纳，才是亲民的应用，才能得到广大用户的喜爱[9]。

关键在与如何将普通用户转化为内内容创造者，内容贡献者总是占用户群体里的很小一部分。但是智能手机和产品体制设定其实都是能让更多的人参与内容贡献的有利因素。

一个可持续可规模化的平台它在早期就是要努力地找到那些内容贡献者，有了内容之后会吸引"普通围观用户"。一旦这群用户达到一定量级后，就得有重要的一步，即尽可能把这群人里的一部分也转换为内容贡献者。如此下来才能有良性循环。

但话又说回来，受众和内容贡献者间也是有鸡-鸡蛋问题。如果没有聚集受众，平台上用户缺少交互，用户贡献内容的欲望也会变小。所以交互也是软件设计中不可或缺的环节，只有信息展示的应用无异于普通的新闻网站，如果加入了用户之间的交流，才会是吸引更多的用户，甚至会使用户在一定范围内以几何倍数增长，且用户忠诚度也会相对很高。

**1.4 图片短视频分享应用的目标**

整个系统分为移动端（Android）和桌面端两个部分，移动端主要实现用户注册，用户登录，用户浏览个人信息，用户段视频及图片，用户分享图片或短视频信息，用户评价关注者信息等，桌面端主要实现了管理员登陆，管理员管理用户，管理员管理用户分享的图片或短视频信息等功能。面对市面上众多雷同的社交应用，本系统要达到以下目标：

1. 系统采用人机交互的方式，界面美观友好，信息上传及查询灵活、方便，数据存储安全可靠。

2. 自动筛选优秀的用户分享向用户推送。

3. 接受用户的反馈，一边对以后功能机界面设计等的改进。

4. 采用模块化的开发方式，使系统应最大限度地实现易维护性和易操作性。

5. 实现分布式计算，减小单台服务器压力，加快请求响应速度。

6. 客户端应用采用跨平台开发技术，使应用可以快速移植到别的平台。

# 二 相关技术分析

## 2.1 B/S软件系统架构

B/S软件系统结构是在服务器（Server）上运行和维护应用，通过客户端的浏览器（Browse）访问应用的一种系统架构，即浏览器/服务器结构。目前Internet流行的Web网站都是一种基于B/S软件系统架构的应用。相对于传统的C/S（Client/Server，客户/服务器）结构，它具有许多独特的优点：

（1）B/S是一种跨平台的、一点对多点及多点对多点的应用软件架构，减少了开发人员在客户端的工作量，使其可以把注意力集中到怎样合理地组织信息、提供客户服务上来。

（2）B/S具有统一的浏览器客户端软件，不仅节省了开发、维护客户端软件的时间与精力，也方便了用户的使用。

（3）B/S架构中客户端只需运行操作系统和Web浏览器，数据的查询、处理和表示都由服务器完成。与C/S架构的应用系统相比，客户端变得非常“瘦”，易于测试和维护。

（4）B/S架构可以透明地跨越异质网络、计算机平台，无缝地联合使用数据库、超文本、多媒体等多种形式的信息。

具体的B/S架构相对C/S架构有几个方面的优势：维护和升级方式简单；成本降低，选择更多；C/S架构的劣势是高昂的维护成本且投资大。

在本次系统开发中，使用基于B/S架构的应用模式和开发技术，很好的满足基于网络环境下的图片共享系统的需求，可以很容易的在互联网上提供服务。

## 2.2 Tomcat简介

TOMCAT是APACHE JAKARTA软件组织的一个子项目，是一个JSP/SERVLET容器，它是在SUN公司的JSWDK（JAVA SERVER WEB DEVELOPMENT KIT）基础上发展起来的一个JSP和SERVLET规范的标准，使用TOMCAT可以体验JSP和SERVLET的最新规范。经过几年的发展，TOMCAT不仅是JSP和SERVLET规范的标准实现，同时具备了很多商业JAVA SERVLET容器的特性，并被一些企业广泛用于商业用途。

Tomcat 运行时，其占用的系统资源小，扩展性好，支持负载平衡与邮件服务等开发应用系统常用的功能；而且它还在不断的改进和完善中，任何一个有能力的程序员都可以更改它或在其中加入新的功能。  
 Tomcat 是一个小型的轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试JSP 程序的首选。对于初学者来说，可以这样认为，当在一台机器上配置好Apache 服务器，可利用它的响应对HTML 页面的访问请求。实际上Tomcat 部分是Apache 服务器的扩展，但它是独立运行的，所以当你运行tomcat 时，它实际上是作为一个与Apache 独立的进程单独运行的。

### 2.2.1 SERVLET容器

负责处理客户的请求。当客户请求来到时，SERVLET容器获取了请求，然后调用某个SERVLET，并把SERVLET的执行结果返回给客户。

当客户请求某个资源时，SERVLET容器使用SERVLETREQUEST对象把客户的请求信息封装起来，然后调用JAVA SERVLET API中定义的SERVLET的一些生命周期方法，最终完成SERVLET的执行，接着把SERVLET执行返回给客户的结果封装在SERVLETRESPONSE对象中，最后SERVLET容器把客户的请求发送给客户，从而完成为客户的一次服务过程。

TOMCAT的默认工作模式是作为独立的SERVLET容器，内置在WEB服务器中的一部分，指使用基于JAVA的WEB服务器的情形。

SERVLET容器作为WEB服务器的插件和JAVA容器的实现。从属WEB服务器的插件在内部地址空间打开一个JVM（JAVA VIRTUAL MACHINE）使JAVA容器得以在内部运行。如某个需要调用SERVLET的请求，插件将获得对此请求的控制，并将它传递（使用JNI）至JAVA容器。进程内的容器对于多线程、单进程的服务器非常适合，而且提供了很好的运行速度，可惜伸缩性有所不足。

注意：JNI是JAVA NATIVE INTERFACE的缩写，为JAVA本地调用接口，通过JNI，JAVA程序可以和其他语言编写的本地程序进行通信。

SERVLET容器是运行于WEB服务器之外的地址空间，并且作为WEB服务器的插件和JVM通过IPC（如TCP/IP）进行通信。进程外容器的反应时间高于进程内的容器，但有较好的伸缩性、稳定性等性能。

IPC INTERPROCESS COMMUNICATION（进程间通信）的简写，它是实现进程间通信的一种技术。

### 2.2.2 TOMCAT的组织结构

TOMCAT是一个基于组件的服务器，构成组件都是可配置的，其中最外层的组件是CATALINA SERVLET容器，其他的组件按照一定的格式要求配置在这个顶层容器中。

TOMCAT的各个组件是在<TOMCAT\_HOME>\conf\server.xml文件中配置的，TOMCAT服务器默认情况下对各种组件都有默认的实现，下面通过分析server.xml文件来理解TOMCAT的各个组件是如何组织的。

<Server> 顶层元素，代表一个服务器

<Service> 顶层元素，是Connector的集合，只有一个Engine

<Connectior/> 连接器类元素，代表通信接口

<Engine> 容器类元素，为特定的Service组件处理所有客户请求，可包含多个Host

<Host> 为特定的虚拟主机处理所有客户请求

<Context> 为特定的WEB应用处理所有客户请求

</Context>

</Host>

</Engine>

</Service>

</Server>

Engine 、Host、 Context是TOMCAT中真正处理客户请求与生成响应的三个组件。

## 2.3 JAVA技术

Java是一门编程语言，也是一个平台，它基于Java虚拟机的技术，借助这种技术建立了跨平台的优势。

Java编程语言与众不同的地方在于：Java程序既是编译型的（转换为一种称为Java字节码的中间语言），又是解释型的（JVM 对字节码进行解析和运行）。编译仅进行一次，而解释在每次运行时都会进行。编译后的字节码采用一种针对JVM 优化过的机器码形式；解释器是JVM 的实现。

Java平台是一种纯软件平台，它有三个版本Java SE、JavaEE、Java ME。

Java SE（Java Platform，Standard Edition）这个版本以前叫做J2SE，后来版本号从1.4直接过度至5.0，其实也就是1.5，所以成为Java SE 5.0。它允许开发和部署在桌面、服务器、嵌入式环境和实时环境中使用的Java 应用程序。Java SE还包含了支持Java Web服务开发的类，并为Java EE这个版本提供基础。

Java EE（Java Platform，Enterprise Edition）这个版本以前叫做J2EE，也就是大家可能听得比较多的，也是Java较流行的领域，Java的企业级版本。Java EE下有很多的流行开发框架，像Struts、Spring、Hibernate，即所谓SSH组合等。

根据项目的大小、应用领域，技术平台的选择还是很多的，除了Java EE，还有.Net，LAMP组合（Linux、Apatch、MySql、PHP组合），相对来说有比较新Ruby在Ruby on Rails的框架应用，后两者可能比较轻量级的Web领域运用成功案例较多，更加重量的企业级潜力还有待挖掘。

Java ME（Java Platform，Micro Edition）这个版本以前叫做J2ME。应用领域为各种移动设备和嵌入式设备，例如：手机、PDA、电视机顶盒和打印机。

Java ME不是太熟悉，不过移动设备和嵌入式设备的开发本人觉得是比较有意思的领域，广为瞩目的就是Google发布的开源智能手机操作系统开发平台AndroID，也就是最近很火的GPhone（原来Google出的不是手机，而是一个手机操作系统开发平台）。这个平台可能会对Java ME有些冲击，按照消息，AndroID包括：1、高度定制的Linux操作系统内核及智能手机硬件驱动支持；2、经过Google修改过的Java虚拟机Dalvik，这里的性能比SUN的Hotspot高，支持大部分Java核心类库；3、大量智能手机开发核心类库；4、大量现成的智能手机应用软件；5、基于Eclipse的开发环境。然而，Java EE虽然提供了统一的编程平台，但不能调用操作系统的资源；而AndroID对下直达操作系统，对上直达应用软件，如：浏览器、日历、地图等。

Java开发工具也有很多，通常使用的IDE有Eclipse和NetBeans。Eclispe比较流行，各类插件也多，使用性起来也不错；NetBeans现在也是越做越好了，这两个IDE的优势主要是开源的，当然还有其它工具如JBuilder、IntelliJ、JCreator。这些IDE各有优势劣势，根据所需开发的项目来选择了。

## 2.4 html5基于android浏览器端开发技术

### 2.4.1 html5技术

HTML5提供了一些新的元素和属性，例如<nav>（网站导航块）和<footer>。这种标签将有利于搜索引擎的索引整理，同时更好的帮助小屏幕装置和视障人士使用，除此之外，还为其他浏览要素提供了新的功能，如<audio>和<video>标记。

（1）取消了一些过时的[HTML4](http://baike.baidu.com/view/1187297.htm" \t "_blank)标记

其中包括纯粹显示效果的标记，如<font>和<center>，它们已经被[CSS](http://baike.baidu.com/view/15916.htm" \t "_blank)取代。

HTML5 吸取了[XHTML](http://baike.baidu.com/view/15906.htm" \t "_blank)2 一些建议，包括一些用来改善文档结构的功能，比如，新的HTML 标签 header, footer, dialog, aside, figure 等的使用，将使内容创作者更加语义地创建文档，之前的开发者在实现这些功能时一般都是使用[div](http://baike.baidu.com/view/26140.htm" \t "_blank)。

（2）将内容和展示分离

b 和 i 标签依然保留，但它们的意义已经和之前有所不同，这些标签的意义只是为了将一段文字标识出来，而不是为了为它们设置粗体或斜体式样。u，font，center，strike 这些标签则被完全去掉了。

（3）一些全新的表单输入对象

包括日期，URL，Email 地址，其它的对象则增加了对非拉丁[字符](http://baike.baidu.com/view/263416.htm" \t "_blank)的支持。HTML5 还引入了微数据，这一使用机器可以识别的标签标注内容的方法，使语义Web 的处理更为简单。总的来说，这些与结构有关的改进使内容创建者可以创建更干净，更容易管理的网页，这样的网页对搜索引擎，对读屏软件等更为友好。

（4）全新的，更合理的Tag

多媒体对象将不再全部绑定在[object](http://baike.baidu.com/view/627051.htm" \t "_blank)或 embed Tag 中，而是视频有视频的Tag，音频有音频的 Tag。

（5）本地数据库

这个功能将内嵌一个本地的SQL 数据库，以加速交互式搜索，缓存以及索引功能。同时，那些离线Web 程序也将因此获益匪浅。不需要插件的丰富动画。

（6）Canvas 对象

将给浏览器带来直接在上面绘制[矢量图](http://baike.baidu.com/view/138039.htm" \t "_blank)的能力，这意味着用户可以脱离Flash 和Silverlight，直接在浏览器中显示图形或动画。

（7）浏览器中的真正程序

将提供 API 实现浏览器内的编辑，拖放，以及各种[图形用户界面](http://baike.baidu.com/view/185360.htm" \t "_blank)的能力。内容修饰Tag 将被剔除，而使用CSS。

（8）Html5取代Flash在移动设备的地位。

（9）其突出的特点就是强化了web页的表现性,追加了本地数据库

## 2.5 MySQL数据库

MySQL是一个小型关系型数据库管理系统，开发者是瑞典MySQL AB公司。在2008年1月16号被Sun公司收购。然而2009年,SUN公司又被Oracal收购.对于Mysql的前途,没有任何人抱乐观的态度。目前MySQL被广泛地应用在Internet上的中小型网站中。由于其体积小、速度快、总体拥有成本低，特别是开放源码这一特点，许多中小型网站为了降低网站总体拥有成本而选择了MySQL作为网站数据库。My SQL有以下特性：

（1）使用C和C++编写，并使用了多种[编译器](http://baike.baidu.com/view/487018.htm" \t "_blank)进行测试，保证源代码的可移植性。

（2）支持AIX、[FreeBSD](http://baike.baidu.com/view/21459.htm" \t "_blank)、HP-UX、[Linux](http://baike.baidu.com/view/1634.htm" \t "_blank)、Mac OS、[Novell](http://baike.baidu.com/view/68826.htm" \t "_blank) Netware、OpenBSD、OS/2 Wrap、Solaris、Windows等多种操作系统 。

（3）为多种编程语言提供了API。这些编程语言包括[C](http://baike.baidu.com/view/10075.htm" \t "_blank)、C++、[Python](http://baike.baidu.com/view/21087.htm" \t "_blank)、Java、[Perl](http://baike.baidu.com/view/46614.htm" \t "_blank)、[PHP](http://baike.baidu.com/view/99.htm" \t "_blank)、Eiffel、Ruby和Tcl等。

（4）支持[多线程](http://baike.baidu.com/view/65706.htm" \t "_blank)，充分利用CPU资源 。

（5）优化的[SQL](http://baike.baidu.com/view/34.htm" \t "_blank)查询算法，有效地提高查询速度。

（6）既能够作为一个单独的应用程序应用在客户端服务器网络环境中，也能够作为一个库而嵌入到其他的软件中提供多语言支持，常见的[编码](http://baike.baidu.com/view/237708.htm" \t "_blank)如中文的GB 2312、BIG5，日文的Shift\_JIS等都可以用作数据表名和数据列名。

（7）提供TCP/IP、ODBC和JDBC等多种数据库连接途径 。

（8）提供用于管理、检查、优化数据库操作的管理工具。

（9）可以处理拥有上千万条记录的大型数据库。

# 三 系统需求分析

## 3.1 需求分析的背景

在完成了针对图片视频分享系统软件市场的前期调查，同时与多位软件使用者进行了全面深入地探讨和分析的基础上，提出了这份软件需求规格说明书。

此需求规格说明书对图片视频分享系统软件做了全面细致的用户需求分析，明确所要开发的软件应具有的功能、性能与界面，使系统分析人员及软件开发人员能清楚地了解用户的需求，并在此基础上进一步提出概要设计说明书和完成后续设计与开发工作。本说明书的预期读者为用户、业务或需求分析人员、测试人员、项目管理人员。

## 

## 3.2 需求分析的任务概述

### 3.2.1 目标

1、开发目标

人们交流的方式多种多样，但是，我们在交流的同时希望更多的了解对方的信息，本软件即是根据此需求进行开发的。

2、应用目标

让用户能够通过注册信息，登录并更新及查询自己的信息，在此基础上，用户还能够有效的掌握和共享其它资源，但是不能更新。从而促进了信息管理的规范化和集成化，使得用户之间的交流更加的便捷。

### 3.2.2 用户的特点

本软件产品的最终用户来源广泛，相同之处则是为了增进与他人之间的交流，管理人员则可以随时更新软件的项目，以及查询和维护信息。本软件面向大众，用户使用过程中的操作也不复杂，因此，本软件的使用频度应达到70%左右。

### 3.2.3 假定和约束

本软件产品为大众用户使用，然而，本软件在功能上还不够完善，因此对于用户的需求还无法完全实现。

## 

## 3.3 系统对主要功能的规定

### 3.3.1 系统功能模块

1. 登录模块
2. 用户注册模块
3. 用户浏览关注人消息
4. 用户浏览系统推荐消息
5. 用户管理关注者与被关注者
6. 用户管理自身信息
7. 管理员管理消息与用户

### 3.3.2 功能描述

下面详细描述一下各个功能模块：

1. 登录模块

仅限于已注册用户和管理人员进行操作。该模块主要是用于用户登录，用户和管理人员输入用户名和正确的密码即可进入对应的界面。该界面的注册控件可转到注册功能，若用户长期未使用已注册帐号（五个月以内未曾使用系统将自动注销）、为注册用户以及用户名或者密码遗失的用户，点击注册控件进入注册模块。

1. 用户注册模块

仅限于用户长期未使用已注册帐号（五个月以内未曾使用系统将自动注销）、为注册用户以及用户名或者密码遗失的用户。在注册过程中，即将注册的用户必须需要根据要求填写用户名、密码、确认密码、注册邮箱等，在填写过程中系统会有对应的提示，\*为必填，对于其他比如说性别、出生年月等信息为可填，注册成功后将进入个人信息模块。

3) 用户浏览关注人消息

每个登陆用户可以查看所有关注人所发布的消息记录以及别的用户对该条消息的评论，以及别的用户对该条消息的赞数量，并且用户可以对每条记录添加评论和赞该条记录。在添加评论页面可以进入用户信息页面。

1. 用户浏览系统推荐消息

系统会自动根据用户喜欢数量来决定向用户推荐消息，每个登陆的用户可以查看系统推荐的消息，对于系统推荐的消息用户可以查看其详细信息：消息发布人，消息评论，消息赞数量，并且可以对消息添加评论，添加赞，并且可以进入用户详细界面。

1. 用户管理关注者与被关注者

每个登陆登陆的用户可以查看自己关注的用户和关注自己的用户，并且可以通过名字来查找用户，进入用户详情页面可以对用户添加关注和取消关注。

1. 用户管理自身信息

每个登陆用户可以对自身信息进行查看和修改，可以修改自身头像，可以删除以往发布的消息。

1. 管理员管理消息与用户

管理员可以查看所有消息和所有用户，并且有权对违规违法用户的消息进行删除和停用其账号。

### 

### 3.3.3 系统主要功能的数据流程图

1. 登陆功能

登陆流程图如图3.3-1所示：

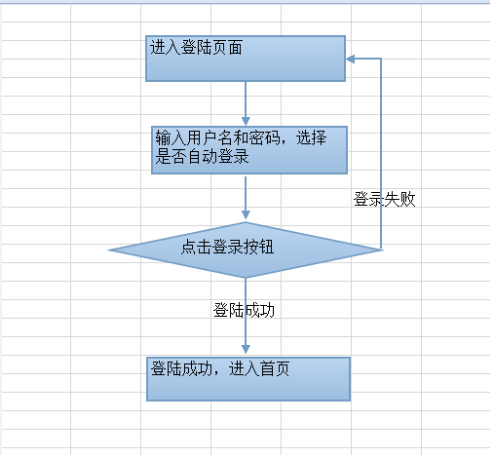


图3.3-1 登陆流程图

1. 用户注册功能

用户注册流程图如图图3.3-2所示：

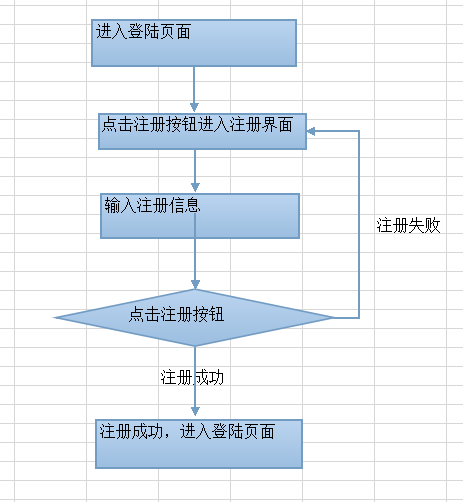


图3.3-2 用户注册流程图

1. 用户浏览关注人消息

用户浏览关注人消息流程图如图3.3-3所示：

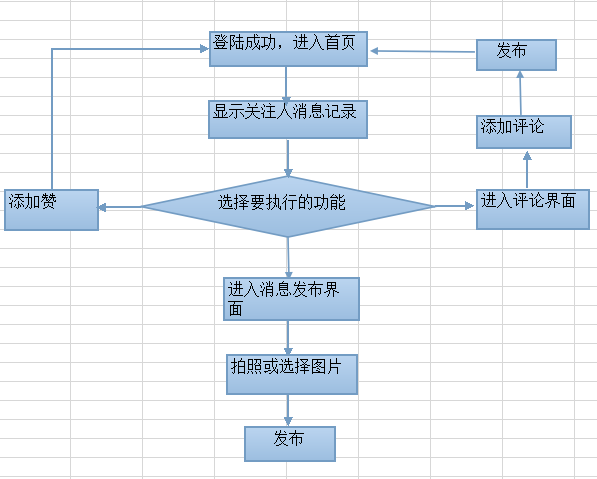
v

图3.3-3 用户浏览关注人消息流程图

1. 用户浏览系统推荐消息

用户浏览系统推荐消息流程图如图3.3-4所示：

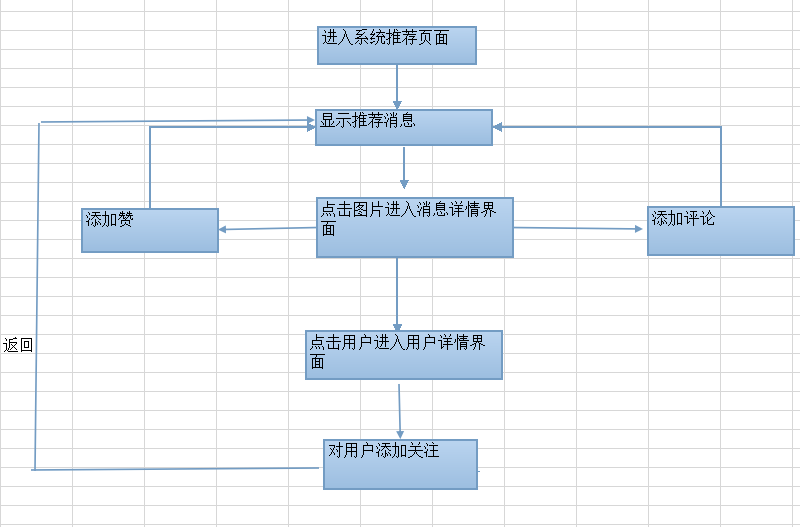


图3.3-4 用户浏览系统推荐消息流程图

1. 用户管理关注者与被关注者

用户管理关注者与被关注者流程图如图3.3-5所示：

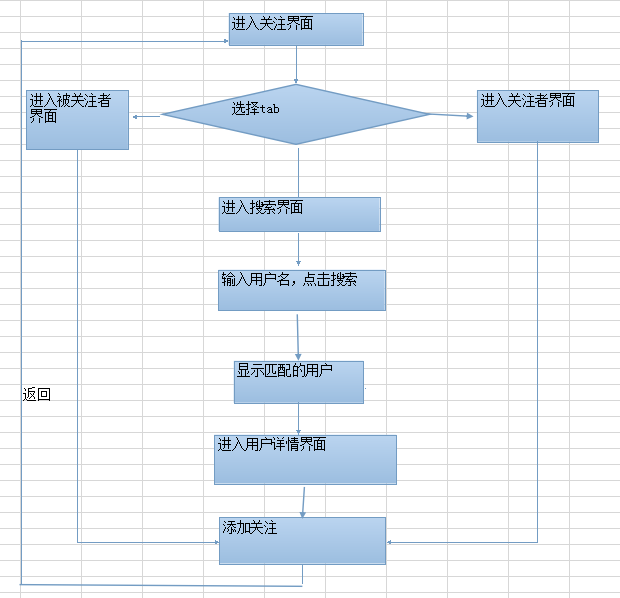


图3.3-5 用户管理关注者与被关注者流程图

1. 用户管理自身信息

用户管理自身信息流程图如图3.3-6所示：

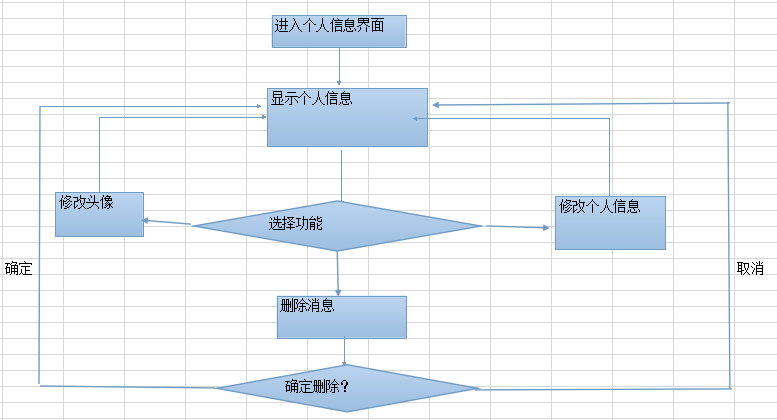


图3.3-6 用户管理自身信息流程图

1. 管理员管理消息与用户

管理员管理消息与用户流程图如图3.3-7所示：

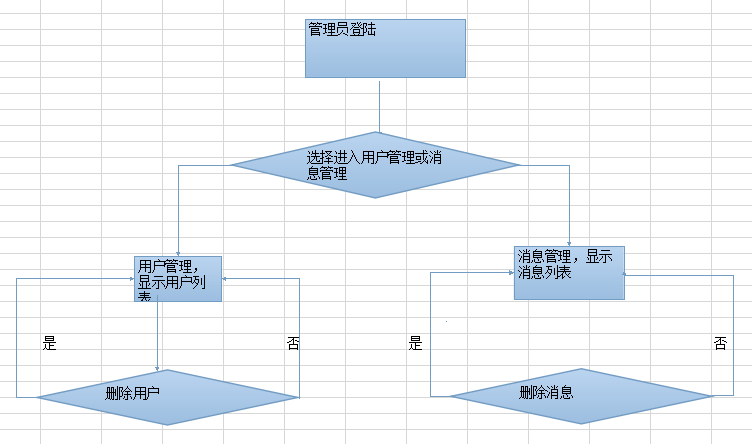


图3.3-7 管理员管理消息与用户流程图

# 四 系统设计与实现

在需求分析通过之后，则进入项目的具体设计阶段，在此阶段需要完成的有详细功能设计，数据库设计，界面设计，软件结构设计，然后编码完成功能实现，最终通过测试之后，交付使用。

## 4.1 系统数据分析

通过需求分析的用例分析，可以得出社团管理系统的数据字典分析和概念设计的实体。

### 4.1.1 系统的数据字典分析

数据字典是用来定义数据流图中各个成分的具体含义，它以一种准确的、无二义性的说明方式为系统的分析、设计及其维护提供了有关元素的一致定义和详细的描述。它与数据流图共同构成了系统的逻辑模型，是需求规格说明书的重要组成部分。数据字典有以下四类：数据流、数据项、数据存储、处理逻辑。数据项是组成数据流和数据存储的最小元素。源点、终点不在系统之内，故一般不在字典中说明。系统数据字典，如表4.1-1所示：

### 4.1.2 数据项的条目

数据项的条目如表4.1-2所示：

### 4.1.3 数据存储条目

数据存储条目表如表4.1-3所示：

表4.1-1 系统的数据字典分析

|  |  |  |  |
| --- | --- | --- | --- |
| 数据流名 | 来源 | 去向 | 说明 |
| 登录 | 用户输入的用户名和密码 | 验证后进入系统界面 | 管理员和用户进入的界面不相同 |
| 用户注册 | 用户填写的注册信息 | 注册成功后可直接登录系统 | 注册信息自动写入到后台数据库 |
| 管理员界面 | 正确的管理员用户名和密码 | 进入主界面进行管理 | 可直接查看用户信息 |
| 用户界面 | 正确的用户名和密码 | 进入用户主界面 | 可对自己的图片视频进行管理 |
| 个人资料 | 用户点击个人档案 | 修改资料或头像 | 修改后的资料自动更新后台数据库 |
| 图片视频浏览 | 用户点击个人图片视频 | 管理自己个人图片视频 | 上传或删除等修改后自动更新后台数据库 |
| 图片视频删除 | 用户进入个人电子图片视频后点击删除图片视频 | 直接删除数据库中的图片视频 | 删除后自动更新后台数据库 |
| 个人电子图片视频 | 用户登录后点击进入 | 对图片视频进行各项管理 |  |
| 资料修改 | 在个人资料中点击 | 进行资料的完善或修改 | 修改后的资料自动更新在数据库中 |

表4.1-2 数据项的条目

|  |  |  |
| --- | --- | --- |
| 数据项名称 | 类型 | 描述 |
| 用户名 | 文本类型 | 描述用户和管理员的用户名 |
| ID | 数字类型 | 描述用户在电子图片视频中的表示 |
| 密码 | 文本类型 | 用户在电子图片视频中的密码 |
| 确认密码 | 文本类型 | 注册时候的确认密码 |
| 注册邮箱 | 文本类型 | 注册时候所用的验证邮箱 |
| 性别 | 文本类型 | 描述用户的性别 |
| 出生年月 | 文本类型 | 描述用户的出生年月 |

表4.1-3 数据存储条目表

|  |  |  |
| --- | --- | --- |
| 数据存储名称 | 简述 | 组成 |
| 用户注册信息 | 存放用户的注册信息 | ID+用户名+密码+确认密码+注册邮箱+性别+出生年月+星座+血型+头像+职业+爱好 |
| 用户图片视频信息 | 存放用户的图片视频信息 | Picture ID+ path |

## 4.2 系统ER模型分析

概念模型是对信息世界的建模，所以概念模型能够方便、准确地表示出信息世界当中的常用概念。概念模型通常是用E-R图来描述现实世界的概念，E-R方法也称为E-R模型。

E-R模型的关键元素是实体、属性和联系。实体是可以在用户的工作环境中标识出的事物，是用户想要跟踪的某个事物。实体在E-R图中，用矩形表示，矩形框内需写明实体名。实体具有属性，有时也称性质，是用来描述实体的特征。属性可以是组合的，或多值的。在E-R图中，属性用椭圆表示，并用无向边将其与相应的实体连接起来。

实体可以通过联系相互关联。在一般的联系中，二元联系居多。二元联系主要有三种类型——l:1，1:n，m: n。在“l:1”联系中，一种类型的单一实体实例与另一个类型的单一实体实例关联。在“l: n”联系中，一个类型的单个实体实例与另一个类型的多个实体实例相关联。在“m: n”联系中，一个类型的单个实例与另一个类型的多个实体实例相关联。同时地，后一类型的单个实体实例与前一类型的多个实体实例相关联。在E—R图中，联系用菱形表示，菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时在无向边旁标上联系的类型（1:1，l: n或者m: n），系统E-R图，如图4.2-1：

管理员

用户

管理

1 N

更新

文档库

更新

I n 1 1

1 N 1 1

图片视频库

更新

更新

图4.2-1 系统E-R

系统实体关系图如图4.2-2所示:

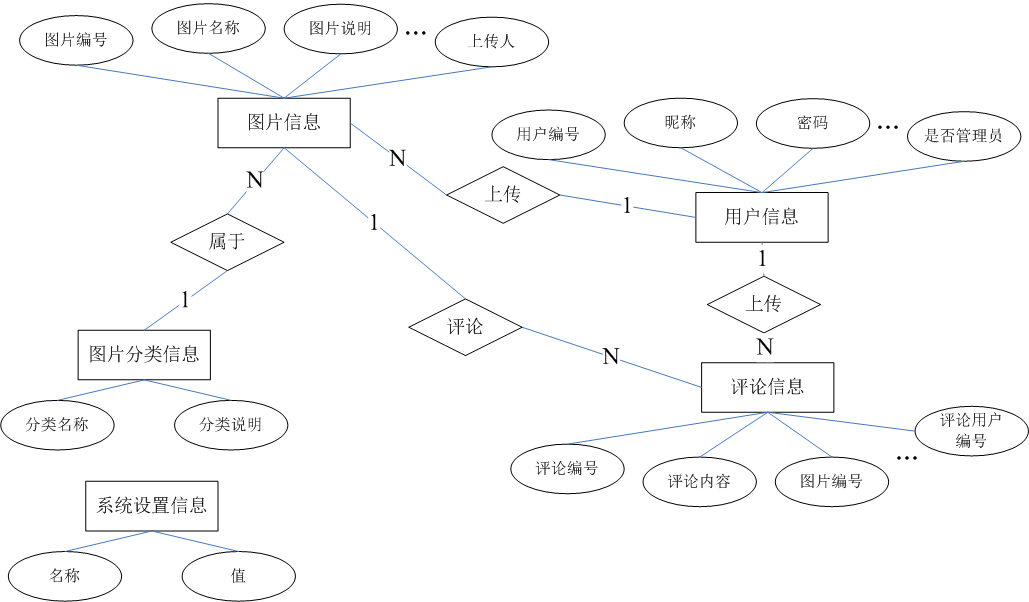
****

图4.2-2 系统实体关系

这里主要有五个实体，包括：图片信息、用户信息、图片分类信息、评论信息和系统设置信息。其中有以下四个实体之间的联系：

（1）图片信息与图片分类信息。一条图片信息对应一条图片分类信息，一条图片分类信息对应多条图片信息。

（2）图片信息与评论信息。一条图片信息可以有多条评论信息对应，但是一条评论信息只对应一条图片信息。

（3）图片信息与用户信息。一条图片信息只能对应一条用户信息，但是一条用户信息可以对应多条图片信息。

（4）评论信息与用户信息。一条评论信息只能对应一条用户信息，但是一条用户信息可以对应多条评论信息。

## 4.3 系统对性能的规定

### 4.3.1 精度

1.要按照严格的数据格式输入，否则系统不给予响应进行处理。

2.查询时要保证查全率，所有相应域包含查询关键字的记录都能查到。

### 4.3.2 时间特性的要求

1. 响应时间：一般操作的响应时间应在1~2秒内，对软磁盘和打印机的操作也应在可接受的时间内完成。
2. 更新处理时间：每12小时
3. 数据的转换和传送时间：
4. 解题时间：

### 4.3.3适应性：

使用开放式的操作系统，开放式的数据库开发工具及标准的通信协议，结构层次自独立性。

## 4.4 系统总体结构

系统总体结构图如图4.4-1所示：

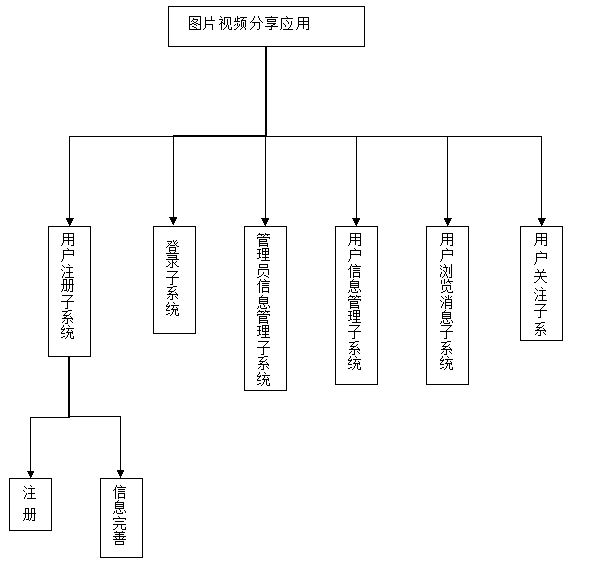


图4.4-1 系统功能结构

## 4.5 数据库设计

根据课题系统需求分析和数据概念模型分析，图片视频分享系统的数据表主要有评论信息表(comment)、关注信息表(follow)、喜欢信息表(like)、消息信息表(message)、角色信息表(role)、用户信息表(userinfo)。具体的数据表结构如下：

评论信息表用来用户评论，评论信息表如表4.5-1所示：

表4.5-1评论信息表（comment表）

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **允许空** | **字段描述** |
| Id | int(auto increment) | 否 | 评论编号（primary key） |
| userId | int | 否 | 评论人id |
| messageId | int | 否 | 被评论消息id |
| Content | Varchar（200） | 否 | 评论内容 |
| Del | Tinyint | 否 | 软删除标志位 |

关注信息表用来保存用户之间的关注信息，关注信息表如表4.5-2所示：

表4.5-2关注信息信息表（follow表）

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **允许空** | **字段描述** |
| userId | Int | 否 | 用户id |
| followerId | Int | 否 | 粉丝id |
| Del | Tinyint | 否 | 软删除标志位 |

喜欢信息表用来保存用户对消息的喜欢信息，喜欢信息表如表4.5-3所示：

表4.5-3喜欢信息表（like表）

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **允许空** | **字段描述** |
| userId | Int | 否 | 粉丝id |
| messageId | Int | 否 | 被喜欢消息id |
| Del | Tinyint | 否 | 软删除标志位 |

消息信息表用来保存用户发布的消息，消息信息表如表4.5-4所示：

表4.5-4消息信息表（message表）

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **允许空** | **字段描述** |
| Id | Int | 否 | 消息id、主键 |
| userId | Int | 否 | 消息发布人id |
| Image | mediumblob | 是 | 图片数据 |
| Video | mediumblob | 是 | 视频数据 |
| videoName | Varchar(200) | 是 | 视频名称 |
| imgOrVideo | tinyInt | 否 | 视频或图片的标志位 |
| Text | Varchar(300） | 是 | 消息文本描述 |
| Del | Tinyint | 否 | 软删除标志位 |

用户角色信息表用来保存系统中用户角色信息，角色信息表如表4.5-6

表4.5-5角色信息表（role表）

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **允许空** | **字段描述** |
| Id | Int(自动增加) | 否 | 角色ID、主键 |
| name | Varchar(10) | 否 | 角色名称 |

用户信息表用来保存用户信息，用户信息表如表4.6-7所示：

表4.5-6用户信息表（userinfo表）

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **字段类型** | **允许空** | **字段描述** |
| Id | Int(auto increment) | 否 | 用户id、主键 |
| Name | Varchar(20) | 否 | 用户真实姓名 |
| Username | Varchar(20) | 否 | 用户昵称 |
| Password | Varchar（20） | 否 | 用户密码 |
| Email | Varchar(50) | 是 | Email |
| Photo | Blob | 是 | 用户头像 |
| Brief | Varchar(50) | 是 | 个性签名 |
| Gender | Tinyint | 否 | 性别 |
| State | Int | 否 | 用户账户状态 |
| Del | Tinyint | 否 | 软删除标志位 |

## 4.6 系统实现

系统实现过程中需要优先实现的是系统实现中的关键问题和难点问题，一般涉及到系统实现的关键处理技术，这些关键技术也是系统设计和实现过程中的亮点，在很大程度上提高了系统的整体质量和开发效率。

### 4.6.1 用户页面访问控制

课题系统主要是实现基于网络环境的图片视频共享，实现中需要解决用户页面访问相关的问题，达到用户只能访问自己能够访问的页面。这里主要实现三方面的访问控制：

（1）图片视频信息浏览模块允许所有用户访问，所以在页面中不需要做限制，用户不需要登录到系统就可以访问。

（2）图片视频管理模块、图片视频评论模块和后台管理模块中的修改密码页面只允许登录到系统上的注册用户访问。

（3）后台管理模块中除了修改密码页面外的页面需要具有管理员权限的用户登录到系统后才能够访问。

用户页面访问控制主要是通过服务器端的Session变量来保存的。对于javaee应用来说，对于每个建立起的会话，会有一组相应的Session变量，当会话终止的时候这些Session变量会被释放，在整个会话中这些Session变量都可以在服务器端的代码中访问，而且对于同一个应用的不同会话，这些Session变量也都是独立的。

用户页面访问控制具体实现的方法如下：

（1）在Global.asax里设置初始Session变量值。

void Session\_Start(object sender, EventArgs e)

{

// 在新会话启动时运行的代码

Session["userid"] = "";

Session["ifadmin"] = "no";

}

Session\_Start函数会在一个会话开始的时候窒息功能，所以在每个会话开始的时候会设置Session变量Session["userid"]和Session["ifadmin"]，分别记录当前登录的用户编号和是否为管理员。

（2）用户登录成功后，会根据用户信息设置变量Session["userid"]和Session["ifadmin"]。

（3）用户注销成功后，会设置Session["userid"]为空串，Session["ifadmin"]为“no”。

（4）在每个需要进行用户访问控制的页面，对这两个Session变量进行检查。具体判断用户类型的流程图如图4.6.1所示。

在判断变量Session["userid"]和Session["ifadmin"]之前都要进行取出两端空格的操作，然后判断Session["userid"]是否为空，如果为空（通过判断字符串长度）则当前没有用户登录；如果不为空，判断Session["ifadmin"]的值是否为“no”，如果是说明登录的用户不是管理员，而是普通注册用户，如果不为“no”则说明登录的是系统的管理员。

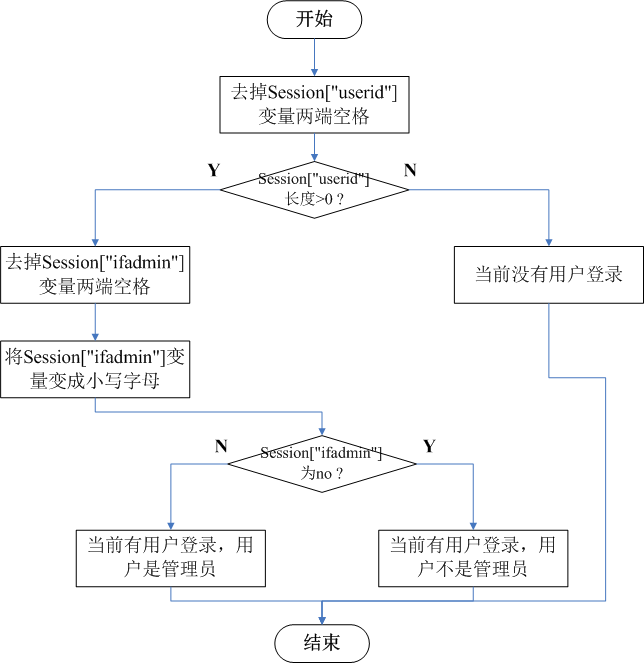


图4.6-1 页面检查用户登录权限的流程

### 4.6.2 在线图片视频浏览

图片视频分享网站的一个很重要的功能就是图片视频文件的上传。在图片视频上传的过程中涉及到两个主要过程：一个是在客户端对用户选择的图片视频进行预览，判断该图片视频是否符合上传要求；一个是通过HTTP协议，将客户端的图片视频上传到服务器上，保存在数据库中。这节主要介绍如何在客户端浏览器上预览图片视频。

实现客户端浏览图片视频的步骤：

1. 主要使用Javascript客户端脚本请求服务器，获取首页json数据，主要代码如下：

**function** getHomePageData(){

showLoading();

$.ajax({

type:"get",

url:"http://192.168.0.102:8088/OnlineShare/viewHomePage?abc="+Math.random(),

dataType : "jsonp",

jsonp: "methodName",

jsonpCallback:"initHomePage",

});

}

1. 服务器端struts2接收到客户端的响应，会调用业务逻辑层的viewFollowingMessage方法，并将请求参数传递给该方法进行处理，该方法会调用持久层方法获取数据库数据并进行处理返回给struts2，struts2对页面做出响应。viewFollowingMessage代码如下：

**public** List<MessageItem> viewFollowingMessage(UserInfo user,Pager pager) **throws** ServiceException

{

List<Message> messages = **new** ArrayList<Message>();

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

**try** {

messages = messageDao.selectFollowingMessages(user, pager);

} **catch** (Exception e) {

e.printStackTrace();

}

}

List<MessageItem> messageList = **new** ArrayList<MessageItem>();

**for** (Message message : messages) {

MessageItem messageItem = ConvertMessageToItem(user, message);

List<Comment> fiveComments = **null**;

**try** {

fiveComments = commentDao.selectFiveCommentsByMessId(message);

} **catch** (DaoException e) {

e.printStackTrace();

}

messageItem.setFiveComments(fiveComments);

messageList.add(messageItem);

}

**return** messageList;

}

（3）客户端接收到服务端返回的json数据进行解析来并显示到界面，客户端方法调用代码如下（核心代码省略）：

**function** initHomePage(message)

{

$(".homePageBody").empty();

showHomePage(message);

}

在线图片视频预览实现后的效果如图4.6.2所示：

### 4.6.3 图片视频上传

图片视频上传功能是图片视频共享网站系统的核心功能之一。将图片视频文件从客户端上传到服务器端保存起来，传输方式和保存方式都有很多种。

图片视频传输可以通过HTTP协议、FTP协议或者P2P共享等方式上传到服务器上。图片视频保存在服务器上可以是在数据库里保存，也可以保存在指定目录中同时在数据库中保存相应的地址。

课题系统中对图片视频上传功能的实现是通过HTTP协议将图片视频从客户端传输到服务器上，然后保存在数据库中。要实现所述的图片视频上传功能，图片上传界面如下图4.6-3所示：

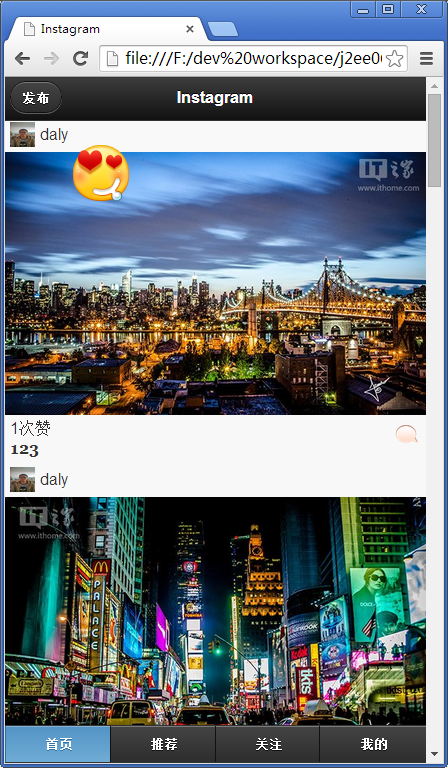


图4.6-2在线图片视频预览的显示

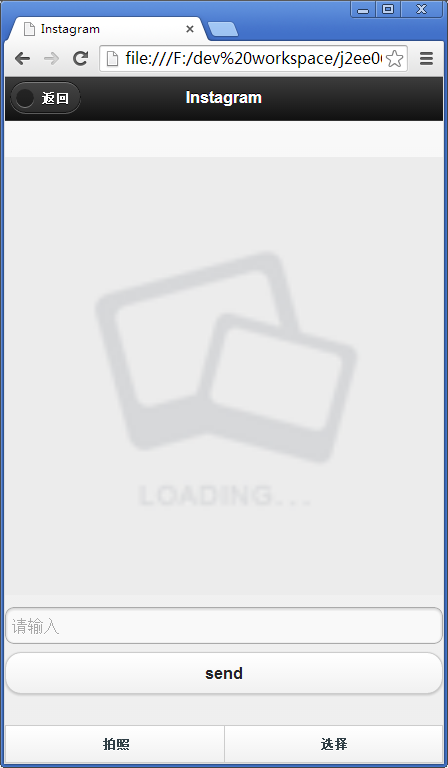


图4.6-3在线图片视频预览的显示

需要做以下几方面的工作：

1. 实例化phonegap的FileTransfer类，设置文件名和文件路径，调用其upload方法，代码如下：

**function** onSub()

{

**var** imgsrc = $("#sendMessage #largeImage").attr("src");

**var** ft = **new** FileTransfer();

**var** options = **new** FileUploadOptions();

options.fileKey="upload";

options.fileName=$(".te").val();

ft.upload(imgsrc,"http://192.168.0.102:8088/OnlineShare/uploadMessage",sendMessageSuccess(),**function**(){},options)

}

（2）服务器端接收方式和使用struts2的upload控件接受方式相同，定义file类型变量，变量名是文件名，即可接受。

（3）struts接收到文件后调用业务逻辑方法将其从file类型转为blob，并调用持久成存入数据库，转换代码如下：

**byte**[] buffer = **new** **byte**[1024 \* 1024];

FileInputStream fis = **null**;

**try** {

fis = **new** FileInputStream(image);

fis.read(buffer);

} **catch** (Exception e) {

e.printStackTrace();

} **finally** {

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

fis.close();

}

}

message.setImage(Hibernate.*createBlob*(buffer));

message.setTime(Timestamp.*valueOf*(sdf.format(date)));

messageService.uploadMessage(message);

}

### 4.6.4 关注模块的实现

关注模块是用户与用户之间关系网搭建的入口，每个用户都可以查看自己的关注人和被关注人，可以按名字搜索用户，并对用户添加关注和取消关注。关注界面如图4.6-4所示：

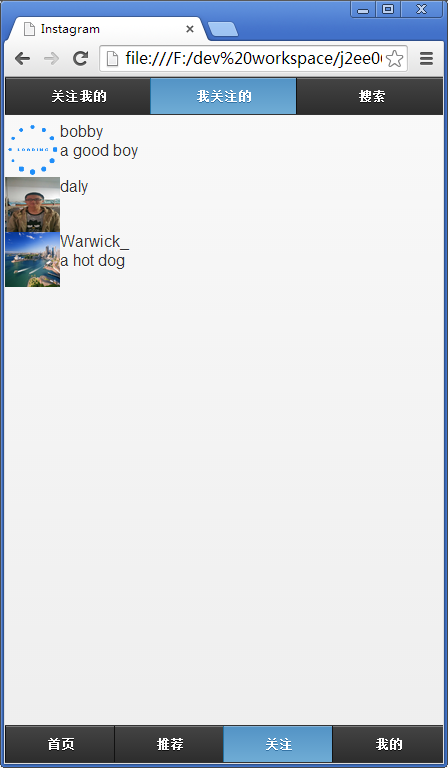


图4.6-4在线图片视频预览的显示

下面简介搜索模块的实现：

1. 客户端接受用户输入数据，并向服务端提交，核心代码如下：

**function** getUserByName(){

showLoading();

$.ajax({

type:"get",

url:"http://192.168.0.102:8088/OnlineShare/getUserByName?id="+Math.random(),

dataType : "jsonp",

data:getQueryString(),

jsonp: "methodName",

jsonpCallback:"showUsers",

});

}

1. 服务端业务逻辑层接收到用户请求后调用持久层从数据库提取数据，并转化为json，核心代码如下所示：

List<UserInfo> userInfos = userService.getUserByName(user,**null**);

List<UserForJson> userForJsons = **new** ArrayList<UserForJson>();

**for** (UserInfo userInfo : userInfos) {

userForJsons.add(convertUserInfo(userInfo));

}

Gson gson = **new** Gson();

String json = gson.toJson(userForJsons);

### 4.6.5 后台管理模块的实现

根据后台管理模块的设计，后台管理模块主要提供用户管理、图片视频分类管理、系统设置信息管理和修改密码的功能。除了修改密码的功能是对网站的注册用户提供的以外，其他部分功能是对网站的管理员提供的，用户必须登录后才能使用相应的功能。

这里介绍一下后台管理模块相关的实现。

（1）注册用户。

该功能为浏览网站的用户提供注册为网站合法用户的功能。

（2）用户管理。

该功能为网站的管理员提供修改用户信息、修改用户密码、删除用户和添加注册用户的功能。首先管理员要先浏览注册用户列表，然后点击相应的链接来修改用户信息、修改用户密码或者删除用户。这些功能需要是网站的管理员登录后才可以使用。如图4.6-5所示：

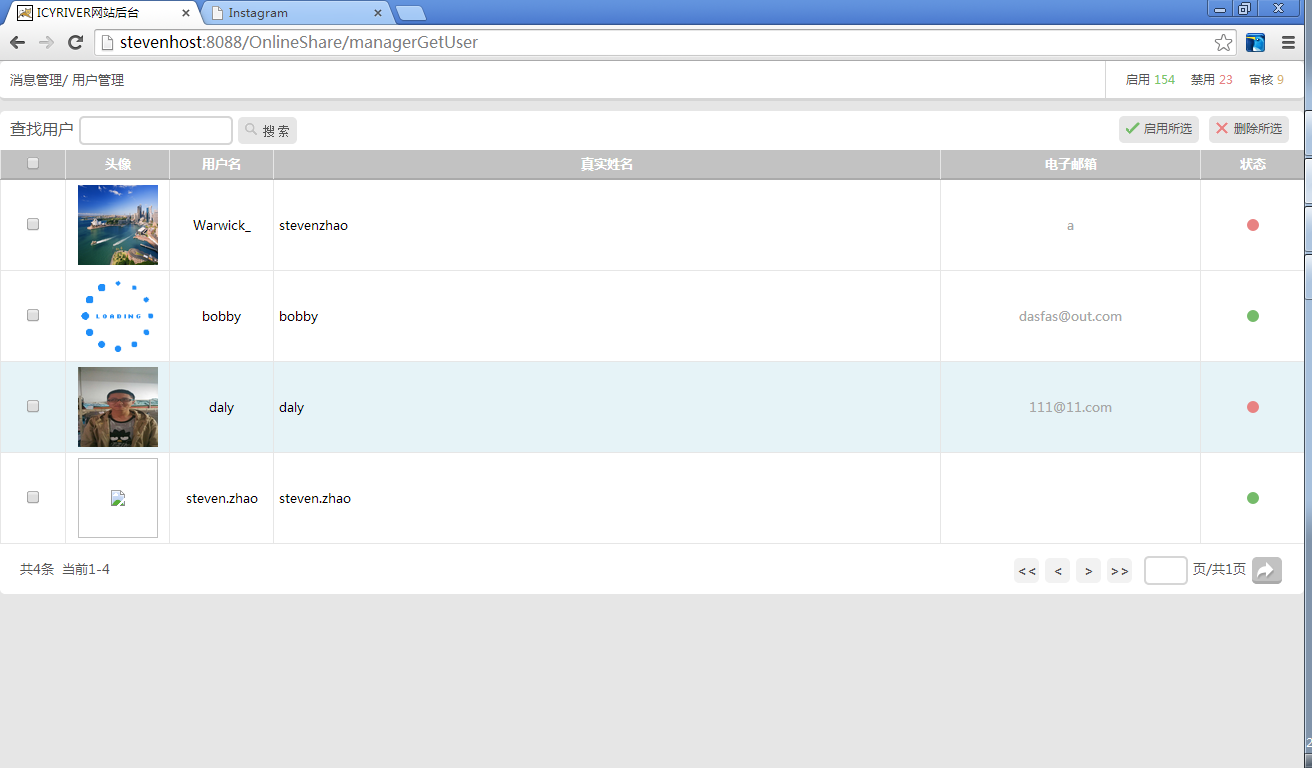


图4.6-5 管理用户的显示

（3）图片视频分类管理。

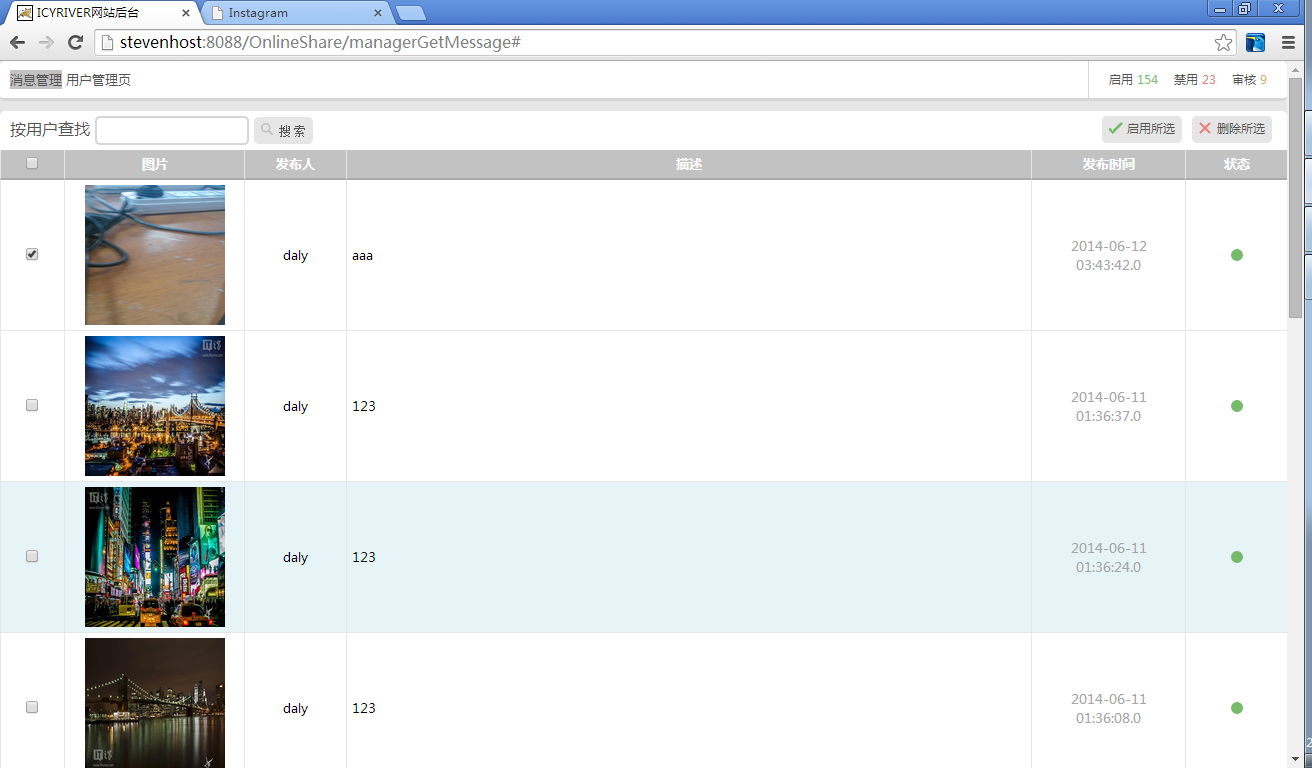
该功能为网站的管理员提供对图片视频分类的管理，包括添加分类、浏览分类、修改分类和删除分类的功能。管理员要先浏览图片视频分类列表，然后点击相应的链接来修改图片视频分类、修改图片视频分类或者删除图片视频分类。这些功能需要是网站的管理员登录后才可以使用。如图4.6-6所示：

图4.6-6管理分类的显示

# 五 系统测试与部署

由于软件开发复杂性的原因，软件开发过程中不可避免的会产生各种错误。为了在软件开发的顺利完成以及在软件投入实际运营之前，尽可能多地发现软件当中的错误，需要尽可能全面地对软件进行测试。软件测试是软件工程中的一个重要阶段，它的任务是确认和验证软件分析、设计、实现过程中的逻辑正确性以及发现软件产品中存在的问题，并通过测试不断提高和完善软件产品的质量。

## 5.1 系统测试概要

此处采用的是黑盒测试法，及主要测试系统的每个功能是否能正确运作。

以下为接受测试的各个功能模块描述：

* 登录功能；

实现用户登录，根据登录者的不同，进入不同界面，主要是管理员界面和用户个人界面。

* 用户注册功能；

实现用户注册功能，在注册时填写相应的用户个人信息（用户名、密码、确认密码等）

* 管理员信息管理功能；

实现管理员对整个电子相册管理系统的管理

* 用户信息管理功能；

实现用户个人信息的管理，用户对于个人信息可以随时的修改。

* 系统设置管理；
* 用户根据需要修改系统相关设置

**5.2 系统功能测试**

表5.2-1用户信息表（userinfo表）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 功能模块 | | 合法等价类 | | 非法等价类 | | 测试结果 |
| 输入 | 预期输出 | 输入 | 预期输出 |
| 登录 | 登录 | 用户名：aa  密码：a | 登录成功 | 用户名：aa密码：a | 用户名或者密码错误 | 成功 |
| 取消 | 不存在 | 关闭 | 不存在 | 关闭 | 成功 |
| 注册 | 不存在 | 用户注册界面 | 不存在 | 不存在 | 成功 |
| 注  册 | 注册 | 用户名：b密码:b  确认密码：b注册邮箱：12334@qq.com | 注册成功 | 输入已存在的用户名aa | 你所注册的用户名已存在，请重新注册 | 成功 |
| 取消 | 不存在 | 退出注册界面 | 不存在 | 不存在 | 成功 |
| 管理员信息管理 | 查看 | 用户名：aa | 个人资料界面 | 不存在 | 不存在 | 成功 |
| 修改 | 用户名：aa | 用户个人资料 | 不存在 | 不存在 | 成功 |

## 5.3 系统维护

本系统主要采取三种方式维护：

1 改正性维护

在软件初期使用，由于某种原因，软件存在一些错误或缺陷。需要由软件开发小组按用户要求对系统进行改正性维护。

2 适应性维护

当系统长时间使用后，随着用户的增加，系统在某些方面的速度可能不如刚开始的时候，所以对这个问题我们要做出适当维护。

3 完善性维护

在系统使用几年后，该应用系统可能不适应业务的发展。用户对软件提出了新的需求和性能要求。

## 5.4 系统部署的软硬件环境

系统部署的软硬件环境如下：

（1）Web服务器 Tomact。

（a）硬件环境：2颗Xeon 3.0G CPU，2G内存，160G硬盘，2块千兆网卡。

（b）软件环境：MySQL，Android。

（2）网络环境：WEB服务器通过网卡100M接入校园网络

## 5.5 系统测试与部署的总体评价

经过系统测试和部署后的用户反馈，对图片共享网站系统的评价如下：

（1）系统实现了需求分析阶段的功能需求，并具有良好的扩展性，能有效地为网络用户提供图片共享服务，系统响应速度较为良好，数据处理效率较高。

（2）经过系统开发阶段的测试和用户测试，表明图片共享网站系统的健壮性比较好，能够有效应对用户对页面的恶意访问，保证系统的稳定运行和后台数据的完整。

（3）系统的人机交互界面较友好，操作流程基本统一。

（4）系统的后台管理功能较完善，能够满足系统日常管理的需要。