**图片管理网站的设计与实现**

摘 要 随着现代互联网和多媒体信息的发展，图片已成为当今互联网信息传播中的一个重要角色。本项目选用运行高效的PHP语言和当今最流行的关系数据库MySQL进行开发，意在为用户提供一个管理、收藏和共享图片的交互平台。本项目主要由用户信息管理、图册管理和图片管理三个部分模块组成，本报告主要讲述了一个图片管理网站从需求分析到系统设计和技术实现以及测试操作的全过程，在细节上分析程序的功能和函数的实现思想。

关键词 图片；管理；分享

**目录**

[1 引用 7](#_Toc23044)

[1.1 课程背景 7](#_Toc12630)

[1.2 课程设计目的 7](#_Toc31019)

[1.3 相关技术简介与开发环境 7](#_Toc12608)

[2 系统结构分析 9](#_Toc12150)

[2.1 系统需求和功能 9](#_Toc483)

[2.2 数据字典 9](#_Toc8732)

[2.3 安全性要求 12](#_Toc16096)

[2.4 一致性要求 12](#_Toc25718)

[2.5 完整性要求 12](#_Toc25470)

[3 系统结构设计 13](#_Toc11451)

[3.1 系统栏目结构图 13](#_Toc31993)

[3.2 系统功能模块设计 13](#_Toc19338)

[3.3 系统流程图 15](#_Toc6249)

[3.4 数据库设计 17](#_Toc25805)

[4系统详细设计与实现 20](#_Toc4234)

[4.1 数据库创建 20](#_Toc7364)

[4.2 数据库备份与恢复 22](#_Toc17207)

[4.3 用户界面设计 22](#_Toc16939)

[5系统测试方案和测试报告 28](#_Toc13145)

[5.1测试方案 28](#_Toc1758)

[5.2测试过程 28](#_Toc8818)

[6 结束语 41](#_Toc22154)

[参考文献 42](#_Toc20524)

[致谢 43](#_Toc19195)

[附录（部分程序代码） 44](#_Toc28250)

1. **引用**
   1. **课程背景**

随着现代计算机和多媒体信息以及数码的普及，当今的信息传播中图片已成为一个重要的元素，数字图片资源已成为我们日常生活中的一个重要部分。我们经常在网络中看到许多自己喜欢的图片，又或希望展示分享自己的图片。把图片存于本地计算机硬盘长期不整理，很难整理分类文件，也存在可能会被删除清空的可能性。而在网络中浏览到的自己喜欢图片，相对零散，很难一次性发现很多自己喜欢的图片。如何更好地存储、分享自己的图片，通过怎样的平台来展示，已经成为了一个问题。

**1.2 课程设计目的**

制作一个具有图片管理、分享和收藏的网站。使用该网站能够帮助用户展示和整理图片，也能帮助用户更容易发现自己喜欢的图片，共享图片资源，已达到互利互赢。但同时，也需注重用户对图片的保留权利，也应增加一切机制和功能来吸引用户的使用，提高用户的积极性。

**1.3 相关技术简介与开发环境**

1. jQuery框架[1]

jQuery是一个快速、简洁的JavaScript框架，是继Prototype之后又一个优秀的JavaScript框架。jQuery设计的宗旨是“write Less，Do More”，即倡导写更少的代码，做更多的事情。它封装JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作、事件处理、动画设计和Ajax交互。

jQuery的核心特性可以总结为：具有独特的链式语法和短小清晰的多功能接口；具有高效灵活的css选择器，并且可对CSS选择器进行扩展；拥有便捷的插件扩展机制和丰富的插件。jQuery兼容各种主流浏览器，如IE 6.0+、FF 1.5+、Safari 2.0+、Opera 9.0+等。

1. PHP[2]

PHP是一种通用开源脚本语言。语法吸收了C语言、Java和Perl的特点，利于学习，使用广泛，主要适用于Web开发领域。PHP 独特的语法混合了C、Java、Perl以及PHP自创的语法。它可以比CGI或者Perl更快速地执行动态网页。用PHP做出的动态页面与其他的编程语言相比，PHP是将程序嵌入到HTML文档中去执行，执行效率比完全生成HTML标记的CGI要高许多；PHP还可以执行编译后代码，编译可以达到加密和优化代码运行，使代码运行更快。

1. MySQL数据库

MySQL数据库简称MySQL，是一款由瑞典MySQL AB公司开发并且广泛应用在于小型企业或者组织中的小型数据管理系统，MySQL数据库因其体积小、速度快、总体拥有成本低受到中小企业的热捧。

1. 开发环境

系统开发环境：Windows 10系统

WEB服务器：Apache 2.0

开发语言：PHP 5.25版

数据库工具：MySQL数据库

开发工具：dreamweaver CC

硬件环境：4G内存win 10系统的笔记本一台

浏览器支持：IE9及IE9以上版本、Firefox 4.0、Chrome 55.0

1. **系统结构分析**

**2.1 系统需求和功能**

1.功能需求

网站需要满足普通用户能方便进行管理图册和图片，通过一些交互操作提高用户使用网站的积极性。管理员能过够对违规用户、图片和图册进行查封隐藏。系统还应具备数据库维护和管理功能等。综合可得本系统对功能的需求如下：

1. 用户的注册与登录。
2. 用户的信息修改。
3. 新建图册
4. 上传图片
5. 修改图册和图片的信息
6. 删除图册和图片
7. 采集图片
8. 其他交互功能：评论、点赞

2.非功能需求

1. 输入输出要求

上传的图片大小在2M以内，且格式限定在JPG,JPEG,BMP,PNG和GIF。

1. 故障处理要求

当数据库连接或其他操作出错，网站能够阻止动作的发生，并给予相应的提示给用户。当用户在未拥有相应权限触发事件，网站能给予相应提示给用户，并提出解决方法。

1. 界面友好性要求

网站的界面设计需美观大方，应该能良好展示已有内容，以吸引用户的注册和使用的积极性。

1. 其他专门要求

为保护原上传者的保留拥有权利，若原作者删除自己上传的图片，所有其他用户有相应图片的记录应该同样被删除。

**2.2 数据字典**

（1）数据结构

用户=用户ID+用户密码+昵称+头像+性别+所在地+注册时间+自我描述

类别=类别ID+类别名称+类别描述

图册=图册ID+所属类别ID+所属用户ID+图册名称+创建时间+图册描述+封面图片ID

图片=图片ID+所属图册ID+图片路径+操作用户ID+操作时间+原图ID+图片描述

评论=评论图片ID+评论用户ID+评论时间+评论内容

1. 数据项

数据项是数据的不可分割的最小单位，根据系统数据结构和对实际情况应用的分析，列出系统的所有数据项，得到如表2-1的数据项表。

表2-1 数据项表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 标识符 | 类型 | 长度 | 所属表名 | 同义名 |
|  | U\_id | 数字 | 16 | User，Album，Photo，Comment，Add\_Like | 用户ID |
|  | U\_nickname | 字符 | 16 | User | 用户昵称 |
|  | U\_psw | 字符 | 16 | User | 用户密码 |
|  | U\_head\_image | 字符 | 65 | User | 用户头像 |
|  | U\_sex | 字符 | 1 | User | 用户性别 |
|  | U\_location | 字符 | 32 | User | 用户所在地 |
|  | U\_reg\_time | 日期 | 19 | User | 用户注册时间 |
|  | U\_des | 字符 | 255 | User | 用户自我描述 |
|  | Ca\_id | 整型 | 4 | Category，Album | 类别ID |
|  | Ca\_name | 字符 | 16 | Category | 类别名称 |
|  | Ca\_des | 字符 | 255 | Category | 类别描述 |
|  | A\_id | 整型 | 4 | Album，photo | 图册ID |
|  | A\_name | 字符 | 16 | Album | 图册名称 |
|  | A\_des | 字符 | 255 | Album | 图册描述 |
|  | A\_time | 日期 | 19 | Album | 图册创建时间 |
|  | A\_cover\_photo | 整型 | 4 | Album，Photo | 封面的图片ID |
|  | P\_id | 整型 | 4 | Photo，Comment，Add\_like | 图片ID |
|  | P\_url | 字符 | 64 | Photo | 图片路径 |
|  | P\_des | 字符 | 255 | Photo | 图片描述 |
|  | P\_from\_photo | 整型 | 4 | Photo | 原图ID |
|  | C\_time | 日期 | 19 | Comment | 评论时间 |
|  | C\_content | 字符 | 255 | Comment | 评论内容 |

1. 数据流

根据所需功能和数据项分析，综合分析数据流动方向，得到如表2-2的数据流表。

表2-2 数据流表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 名称 | 来源 | 去向 | 组成 |
| 1 | 身份信息 | 系统用户 | 应用系统 | 用户姓名+密码 |
| 2 | 授权信息 | 应用系统 | 系统用户 |  |
| 3 | 错误身份信息 | 应用系统 | 系统用户 |  |
| 4 | 删除请求 | 系统用户 | 应用系统 | 待删除的记录信息 |
| 5 | 删除结果 | 应用系统 | 系统用户 | 删除后的结果信息 |
| 6 | 修改请求 | 系统用户 | 应用系统 | 待修改的记录信息 |
| 7 | 修改结果 | 应用系统 | 系统用户 | 修改后的结果信息 |
| 8 | 添加请求 | 系统用户 | 应用系统 | 待添加的评论信息 |
| 9 | 添加结果 | 应用系统 | 系统用户 | 所添加的评论信息 |
| 10 | 搜索请求 | 系统用户 | 应用系统 | 待查询的对象模糊信息 |
| 11 | 搜索结果 | 应用系统 | 系统用户 | 被查询对象具体信息 |

**2.3 安全性要求**

为了防止非法用户对服务器和数据库进行非法操作，必须登录再操作，而没有账号的用户必须进行注册拥有账号。用户登录密码也需要被加密保护。数据来源于用户输入存储，所以也应该考虑把特殊的代码字符加以标记注释保存调用，以防对网站进行破坏。

**2.4 一致性要求**

实体以用户、图册、图片为中心，之间具有较强的联系。由此数据表之间也有较强的联系，图册依赖于分类和用户，图片依赖于用户和图册。为了更好维护关系，需要设计外键了来保持一致性[2]。

**2.5 完整性要求**

数据库的完整性是指数据库的正确性，有效性和相容性。数据库的信息都由用户输入，而数据的输入由于种种原因，会发生输入无效或者错误信息的情况。应该有几重设置筛选保证数据录入的正确有效，防止出现必须信息内容为空。

1. **系统结构设计**

**3.1 系统栏目结构图**

为了能更好呈现网站的内容，网站设置了几个主要栏目来展示不同的信息。首页内容分为展示热门图片、热门图册和全部分类。用户的个人主页包括了操作图片和图册的信息展示和资料修改页面。图片和图册列表页面包括全站图片、图册显示和分类显示。详细网站的栏目结构图如图3.1所示。

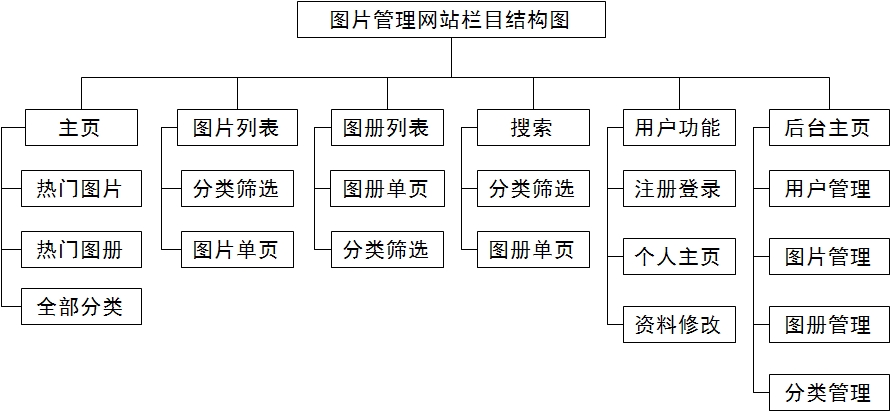
****

图3.1 图片管理网站栏目结构图

**3.2 系统功能模块设计**

本系统的设计初步规划为可以三个基本模块，分别是用户模块、图册模块和图片模块，图3.2展示了系统模块的层次图，三个基本模块的功能可划分为下：

（1）用户模块

注册：用户可注册成为会员，拥有个人页面。

登录：通过用户帐号和密码进行验证登录。

修改个人信息：自定义用户的个人信息。

（2）图册模块

新建：通过输入图册标题、图册描述并选择图册分类进行新建。

删除：通过选中指定图册对其进行删除。

修改：通过选中指定图册对其信息进行修改。

查询：通过输入关键字进行模糊信息查询。

设置封面：对图册进行封面的设置。

（3）图片模块

添加：通过输入图片描述、和所属图册进行添加。

删除：在上传图片中选择指定图片进行删除。

修改：在上传图片中选择制定图片对其信息进行修改。

查询：通过输入关键字进行模糊信息查询。

采集：收藏他人发布上传的图片。

喜欢：喜欢他人发布上传的图片。

评论：对图片进行评论。

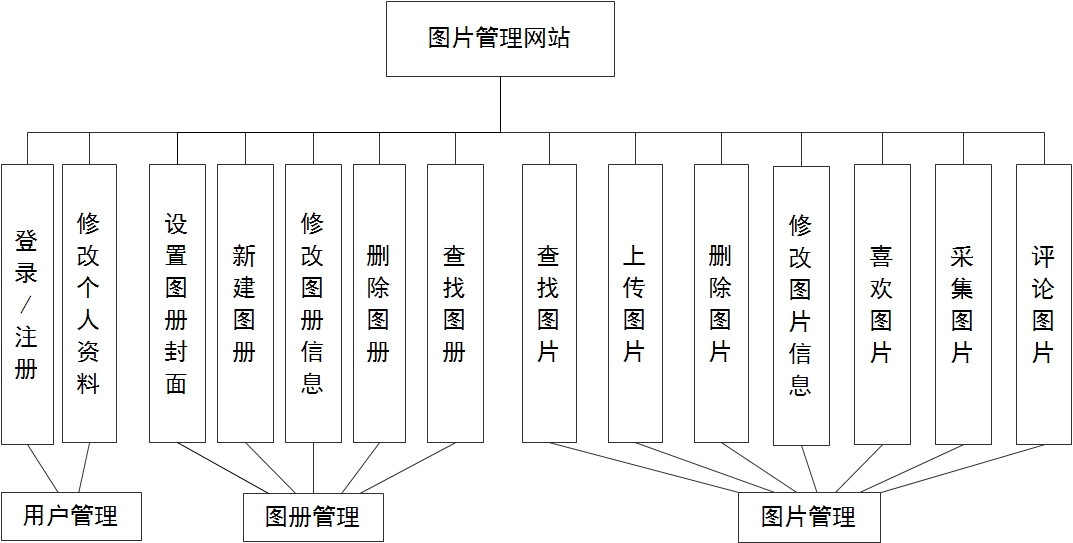


图3.2 图片管理网站基本功能图

**3.3 系统流程图**

本系统在进行图片管理和图册管理操作前，必须要进行登录操作才可以执行，若没有账号则需要进行注册再登录。

如图3.3是整个操作流程的过程。用户在登录后可选择管理自己的个人信息、进行对图册或图片的操作。图3.4则对整体操作流程中的图册操作流程进行了详细描绘，图3.5对整体操作流程中的图片操作过程进行了描绘。

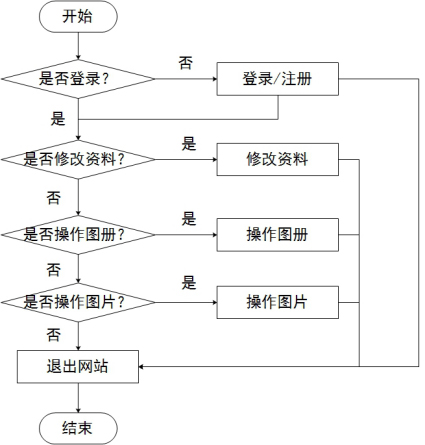


图3.3 总体系统流程图

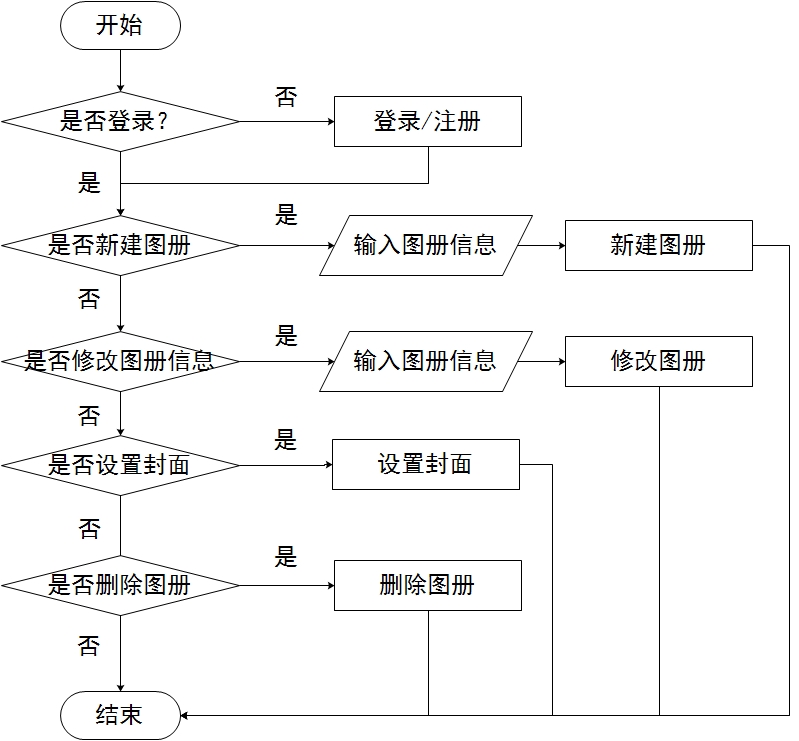


图3.4 图册操作流程图

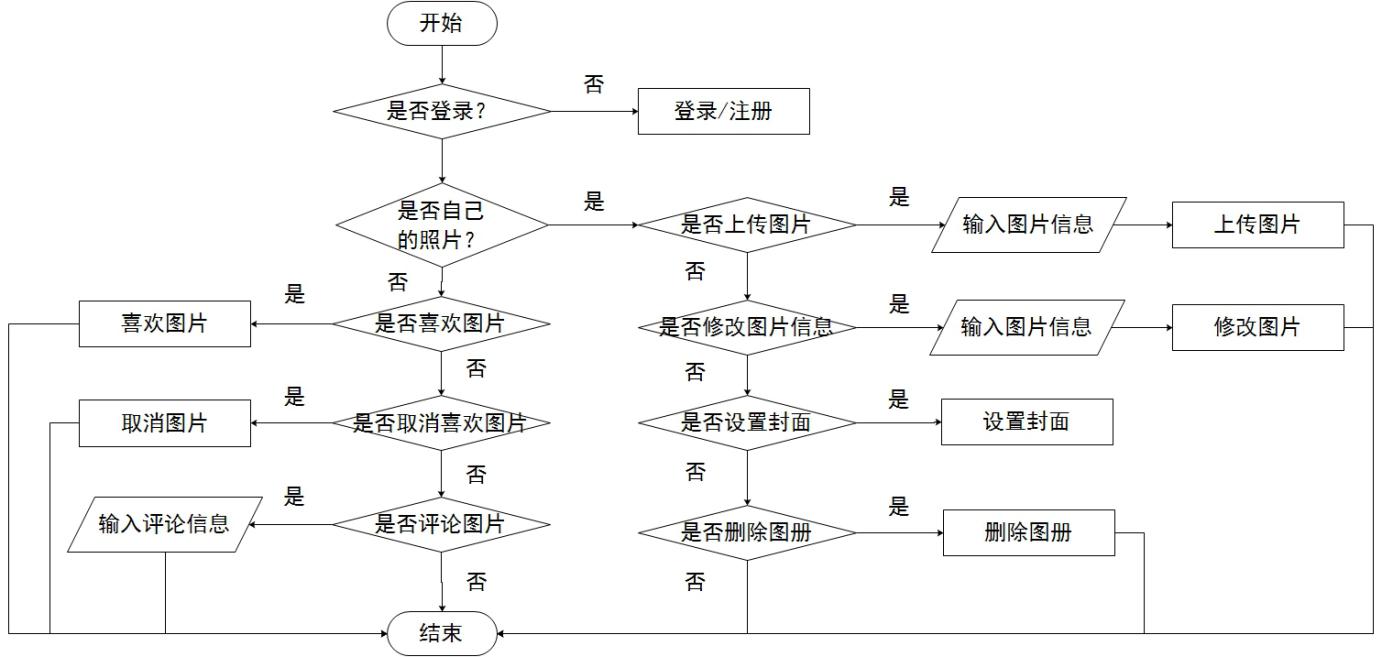


图3.5 图片操作流程图

**3.4 数据库设计**

1.系统E-R图

本网站系统的主要实体有用户、图册和图片，它们之间存在着紧密的实体联系。通过对子系统的局部E-R图设计，消除冗余和不一致，最终得到以下图3.3图片管理网站的主E-R图

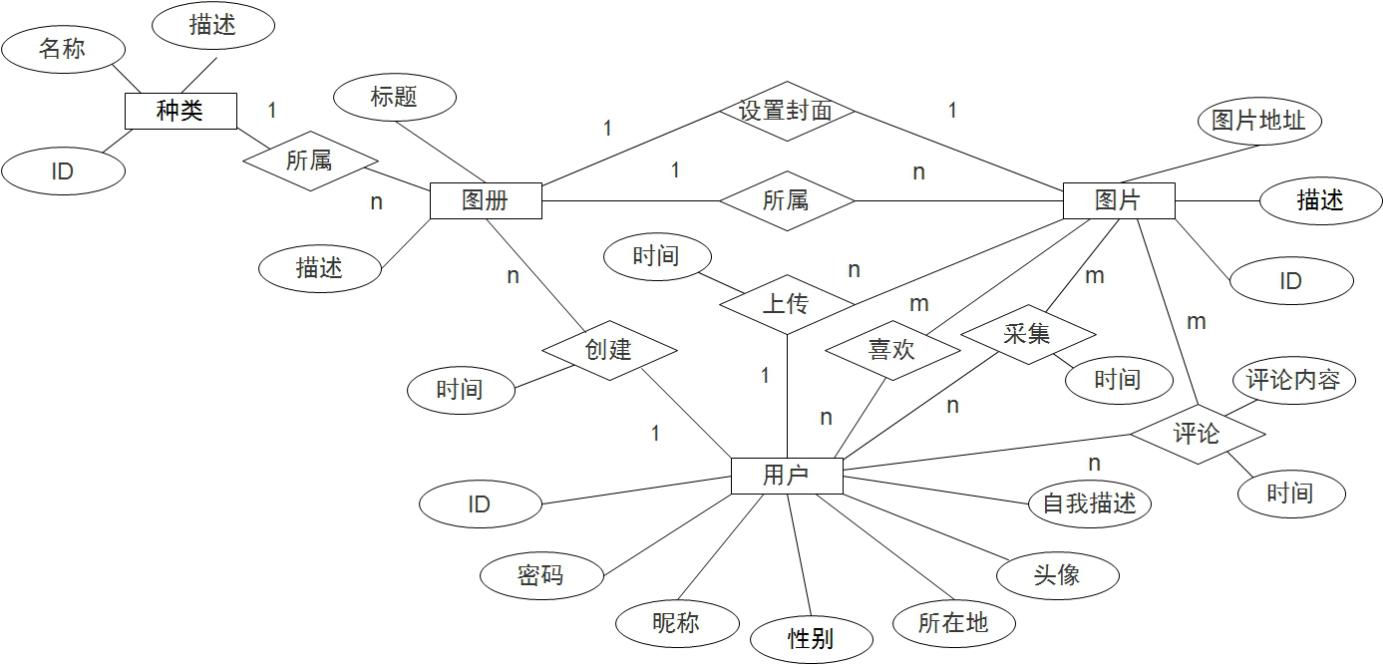


图3.6 图片管理网站主E-R图

1. 逻辑结构设计

逻辑结构设计的主要目的就是产生一个具体DBMS可处理的数据模型和数据库模式，按照以下三步进行分析。

（1）关系模式设计

根据概念结构设计，可以将E-R图转化为以下的关系模式：

User(u\_id,u\_psw,u\_nickname,u\_head\_image,u\_sex,u\_reg\_time,u\_location,u\_des,

u\_status);

Categoty(ca\_id,ca\_name,ca\_des);

Album(a\_id,a\_name,ca\_id,a\_time,a\_des,u\_id,a\_cover\_photo);

Foreign key(u\_id,a\_cover\_photo)

Photo(p\_id,p\_time,p\_url,p\_des,u\_id,a\_id,p\_from\_photo)

Foreign key(u\_id,a\_id,p\_from\_photo)

Comment(p\_id,u\_id,c\_time,c\_content);

Foreign key(p\_id,u\_id)

Add\_like(p\_id,u\_id);

Foreign key(p\_id,u\_id)

（2）优化分析

表User(u\_id,u\_psw,u\_nickname,u\_head\_image,u\_sex,u\_reg\_time,u\_location,u\_des,

u\_status)，其中u\_id是主键，其他非主属性完全依赖与主属性，属于BC范式，消除了各种插入和删除异常。

表Categoty(ca\_id,ca\_name,ca\_des)，其中ca\_id是主键，此表是图册表通过范式分离而达到BC范式的表，设置编号为主键，避免删除异常和大片数据的更改。

表Album(a\_id,a\_name,ca\_id,a\_time,a\_des,u\_id,a\_cover\_photo)，其中a\_id是主键，其他非主属性完全依赖与主属性，属于BC范式。

表Photo(p\_id,p\_time,p\_url,p\_des,u\_id,a\_id,p\_from\_photo)，其中p\_id是主键，其他非主属性完全依赖与主属性，属于BC范式。

表Comment(p\_id,u\_id,c\_time,c\_content)选用p\_id、u\_id和c\_time三个组合成为主键，能唯一确定内容，也属于BC范式。

而表Add\_Like的设计结果显而易见，也属于BC范式。对于六张表都是BC范式，数据插入，删除，修改都具有较强的处理功能，也比价适合数据库的存储。

（3）系统完整性设计

User(u\_id,u\_psw,u\_nickname,u\_head\_image,u\_sex,u\_reg\_time,u\_location,u\_des,

u\_status);其中u\_id是主键，u\_psw，u\_nickname，u\_reg\_time约束为非空，u\_sex的约束为’男’或’女’，默认为’男’。

Categoty(ca\_id,ca\_name,ca\_des);其中ca\_id是主键，ca\_name约束为非空。

Album(a\_id,a\_name,a\_time,a\_des,ca\_id,u\_id,a\_cover\_photo);其中a\_id是主键，ca\_id、u\_id、a\_cover\_photo是外码，分别参照category表的ca\_id、user表的u\_id和photo表的p\_id。a\_name、a\_time约束为非空。

Photo(p\_id,p\_time,p\_url,p\_des,u\_id,a\_id,p\_from\_photo);在这其中p\_id是表的主键，而u\_id、a\_id和p\_from\_photo是外码，分别参照了user表的u\_id，album表的a\_id和photo的p\_id。

Comment(p\_id,u\_id,c\_time,c\_content);p\_id、u\_id和c\_time三个组合成为主键，而p\_id、u\_id分别参照photo表的p\_id和user表的u\_id。

Add\_Like(u\_id,p\_id);其中u\_id和p\_id是主键，而u\_id和p\_id分别参照user表和photo表。

4系统详细设计与实现

**4.1 数据库创建**

（1）各数据库表说明

综合以上分析，加以数据库中基本范式要求分解，完成了数据库设计。其中表4-1汇总了所包括的全部数据表，阐述了各个表的内容。而表4-2，表4-3，表4-4，表4-5，表4-6和表4-7则分别说明了各个表的具体设计。

表4-1 数据表描述

|  |  |  |
| --- | --- | --- |
| 编号 | 数据表名称 | 内容 |
| 01 | User | 记录用户的资料与信息 |
| 02 | Category | 记录类别 |
| 03 | Album | 记录图册信息 |
| 04 | Photo | 记录图片信息 |
| 05 | Comment | 记录评论的信息 |
| 06 | Add\_Like | 记录用户对图片的“喜欢”操作 |

表4-2 用户表(user)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 属性 | 说明 |
| u\_id | varchar | 16 | 主码 | 用户ID |
| u\_psw | char | 40 |  | 登录密码 |
| u\_nickname | varchar | 16 |  | 昵称 |
| u\_head\_image | varchar | 64 |  | 头像图片地址 |
| u\_sex | char | 1 |  | 用户性别 |
| u\_reg\_time | datetime | 19 |  | 注册时间 |
| u\_location | varchar | 30 |  | 所在地 |
| u\_des | varchar | 255 |  | 自我描述 |

表4-3 图册表(album)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 属性 | 说明 |
| a\_id | int | 16 | 主码 | 相册ID |
| a\_name | char | 40 |  | 相册标题 |
| u\_id | varchar | 16 | 外码 | 创建人ID |
| ca\_id | int | 4 | 外码 | 分类所属ID |
| a\_time | datetime | 19 |  | 创建时间 |
| a\_des | varchar | 255 |  | 描述 |
| a\_cover\_photo | int | 4 | 外码 | 封面图片ID |

表4-4 图片表(photo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 属性 | 说明 |
| p\_id | int | 16 | 主码 | 图片ID |
| u\_id | int | 4 | 外码 | 采集/上传人ID |
| a\_id | int | 4 | 外码 | 所属图册ID |
| p\_url | varchar | 64 |  | 图片路径 |
| p\_time | datetime | 19 |  | 上传/采集时间 |
| p\_des | varchar | 255 |  | 图片描述 |
| p\_from\_photo | int | 4 | 外码 | 采集图片来源ID |

表4-5 分类表(category)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 属性 | 说明 |
| ca\_id | int | 4 | 主码 | 分类ID |
| ca\_name | varchar | 16 |  | 分类名称 |
| ca\_des | varchar | 255 |  | 分类描述或别名 |

表4-6 评论表(comment)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 属性 | 说明 |
| p\_id | int | 4 | 主码 | 评论图片ID |
| u\_id | varchar | 16 | 主码，外码 | 评论用户 |
| c\_time | datetime | 19 |  | 评论时间名 |
| c\_content | varchar | 255 |  | 评论内容 |

表4-7 喜欢表(add\_like)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 属性 | 说明 |
| p\_id | int | 4 | 主码，外码 | 被喜欢图片ID |
| u\_id | varchar | 16 | 主码，外码 | 喜欢用户 |

**4.2 数据库备份与恢复**

（1）mysqldump命令备份与恢复[3]

mysqldump命令将数据库中的数据备份成一个文本文件。表的结构和表中的数据将存储在生成的文本文件中。mysqldump命令的工作原理很简单。它先查出需要备份的表的结构，再在文本文件中生成一个CREATE语句。然后，将表中的所有记录转换成一条INSERT语句。然后通过这些语句，就能够创建表并插入数据。

还原使用mysqldump命令备份的数据库的语法如下：mysql -u root -p [dbname] < backup.sq，比如：mysql -u root -p < C:\backup.sql

（2）复制目录实现备份与恢复

MySQL可以把的数据库文件直接复制出来备份。这是最简单，速度最快的方法，但是在此之前，要先将服务器停止，这样才可以保证在复制期间数据库的数据不会发生变化。而且这种方法不适用于InnoDB存储引擎的表，对于MyISAM存储引擎的表很方便。恢复时必须必须保证两个MySQL数据库的版本号是相同的[4]。

**4.3 用户界面设计**

1. 首页

网站的首页主要是展示热门图片和推荐图册，希望能够做到吸引网站游客浏览和注册。如图4.1所示则是本网站的首页。

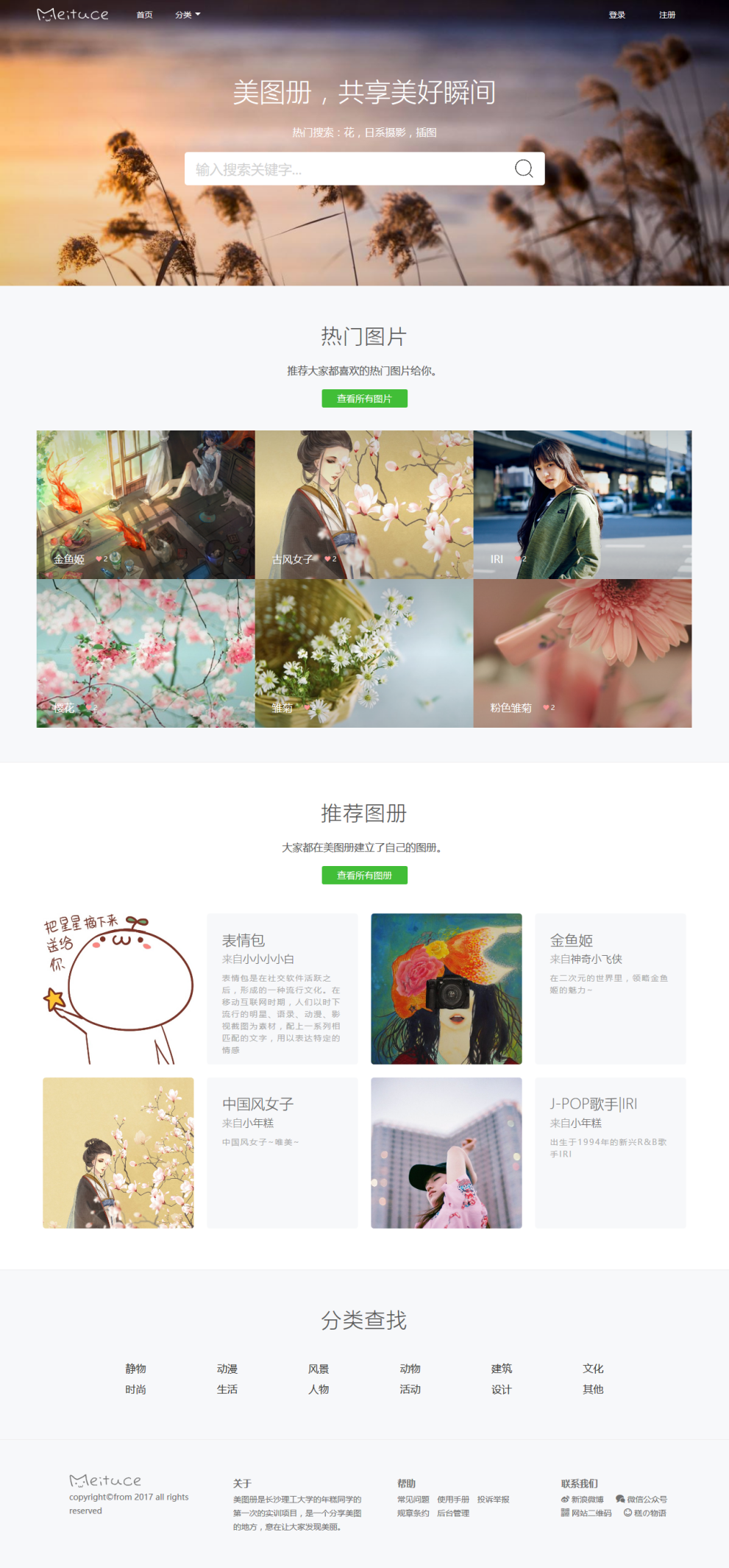
****

图4.1 首页设计

1. 登录/注册

用户在登录网页之后，没有账号的用户可以点击注册页面，如图4.2进行注册，注册之后自动进入登录后页面。而已拥有登录账号的用户在登录页面，如图4.3输入用户名和密码方可实现登录。

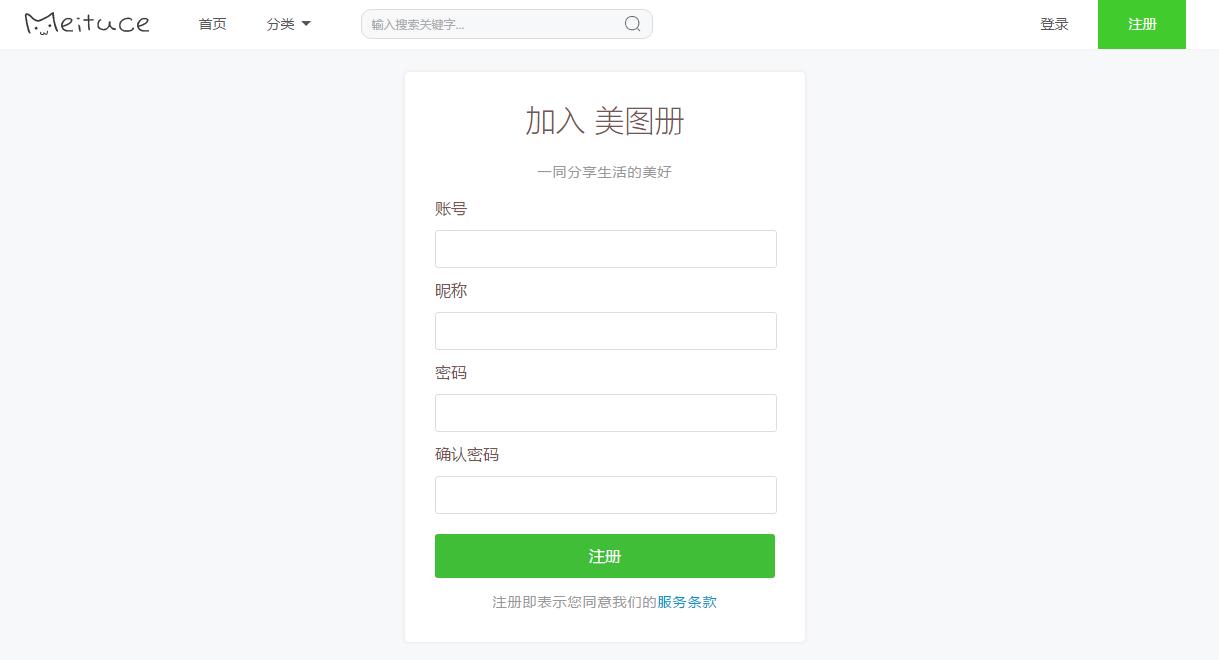


图4.2 用户注册页面

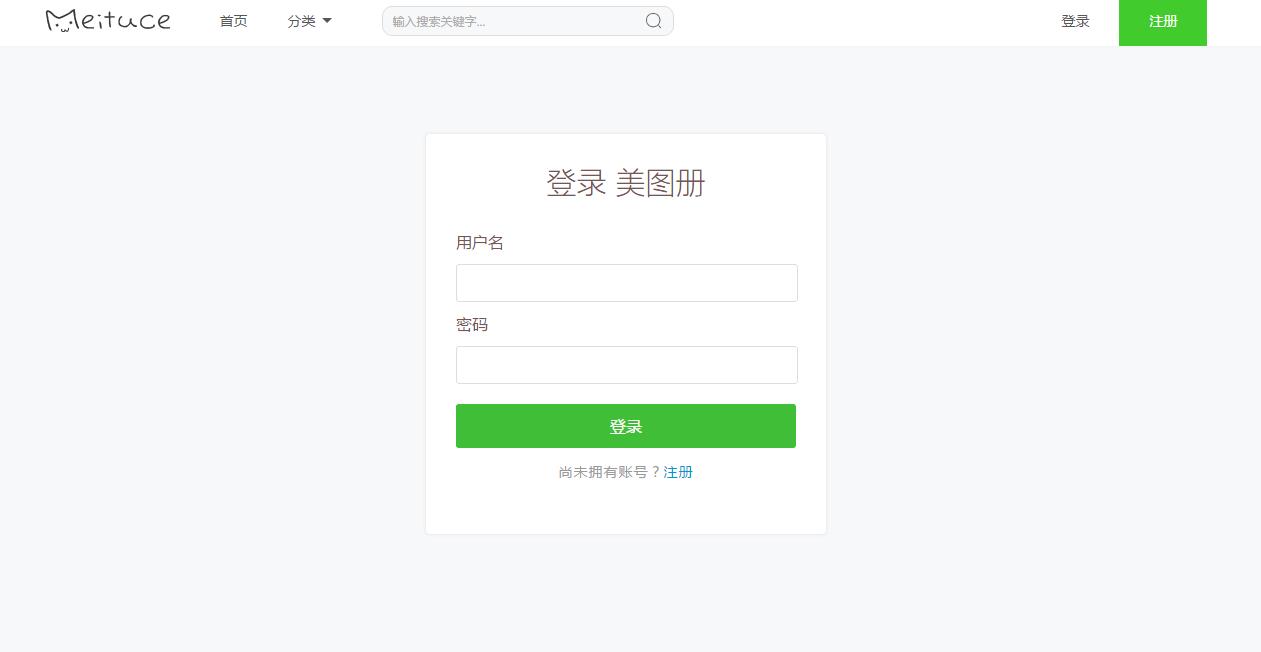


图4.3 用户登录页面

1. 用户主页

用户在登录后可进入用户个人主页，可以浏览用户所有创建的图册和上传、采集的图片，可以进行个人信息设置、上传图片和新建图册的操作，如图4.4所示。



图4.4 个人主页页面

用户进行信息更改，相应输入规范的提示都会显示在右侧的灰色小字里。若不符合要求，则会进行相应的提示，不给予提交，如下图4.5所示。

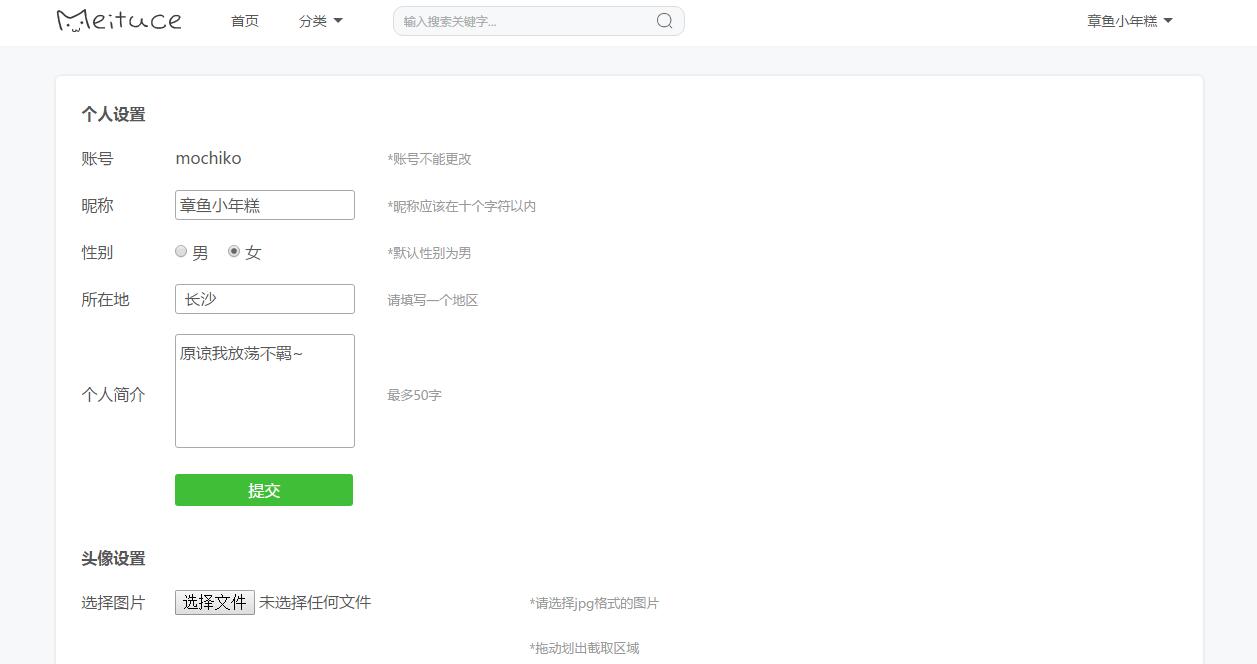


图4.5 信息修改页面

1. 图册内页

在图册内页可以实现删除图册、编辑图册和设置图册封面的功能，如图4.6所示。



图4.6 图册内页

1. 图片内页

在用户自己上传的图片的图片内页可以实现删除图片、编辑图片、评论图片和查看大图。若是其他用户上传的图片界面则可以采集图片、喜欢图片、评论图片和查看大图。如图4.7所示。

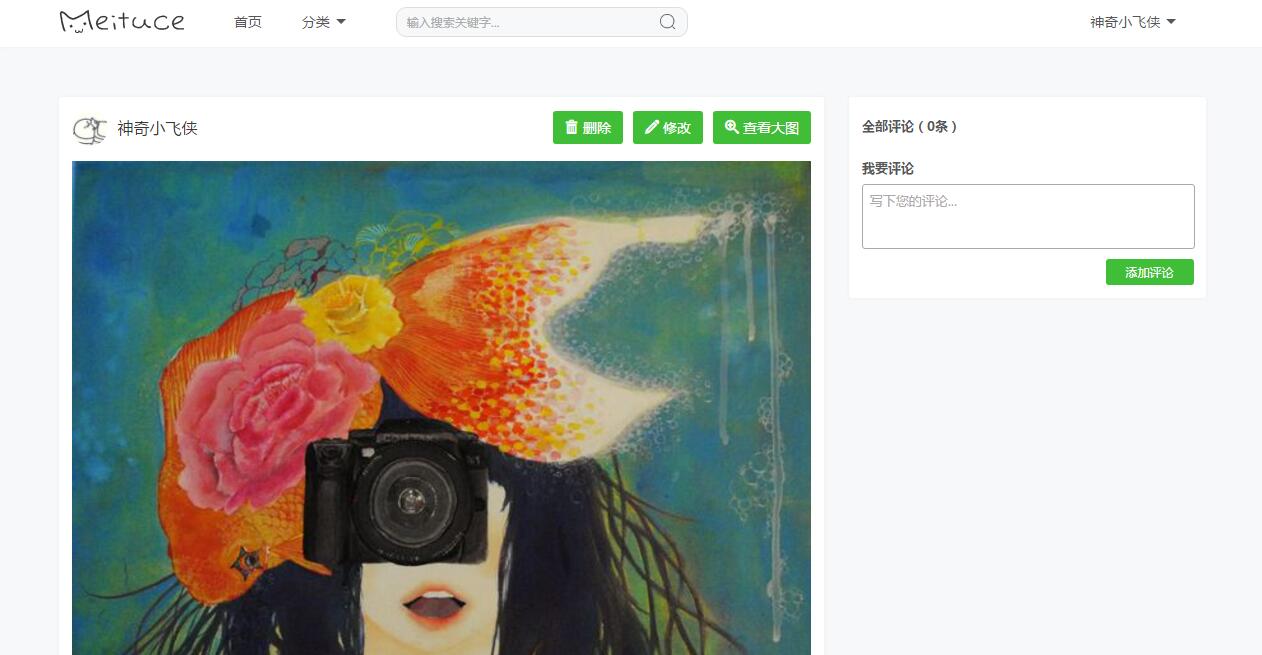


图4.7 图片页面

1. 搜索页面

用户可在头部搜索栏进行搜索，结果将会从图片的描述、所属图册的描述、分类和分类描述中的关键词作为搜索参考。用户可以对图片和图册两种对象进行筛选，如图4.8所示。

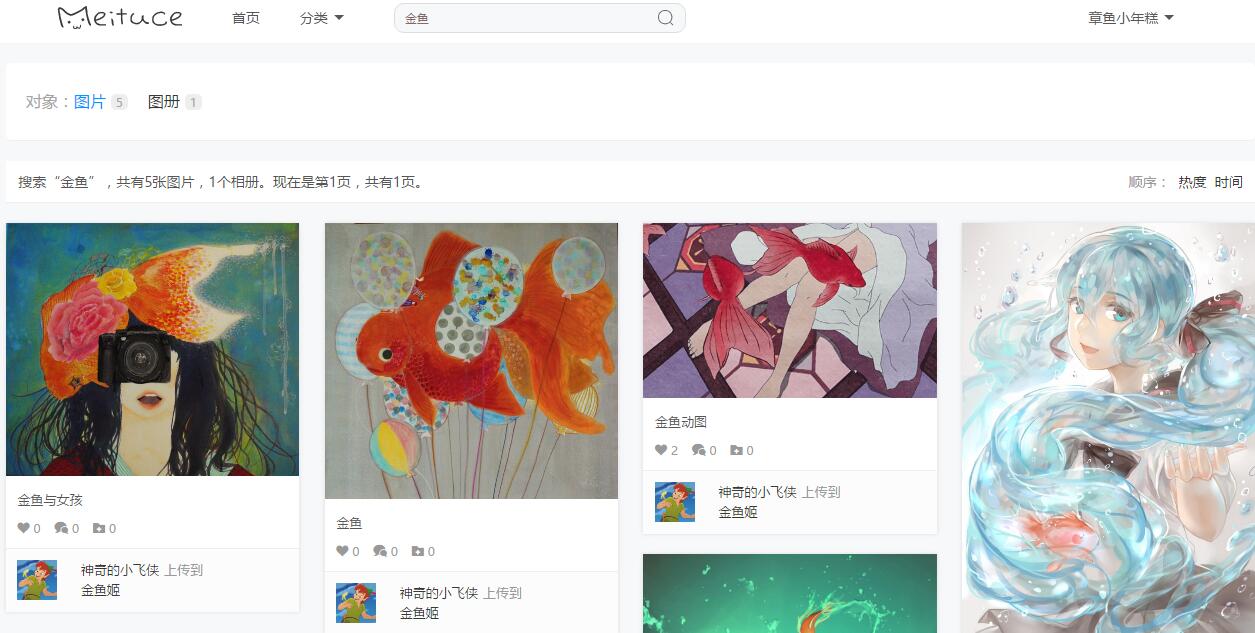


图4.8 搜索结果

5系统测试方案和测试报告

5.1测试方案

由于系统规模不大，所以没有安排很复杂的测试过程。整个测试过程基于自顶向下测试的组装模块的方法，先对主模块进行基本测试，然后在按深度优先策略逐一将子模块组装到主模块上进行测试，最后再对系统进行全面的整体测试。

5.2测试过程

1. 用户记录测试
2. 注册

如图5.1，图5.2和图5.3所示，正确填写信息会显示“注册成功”弹窗。若填写错误则有相应的提示信息。



图5.1 正确填写表单

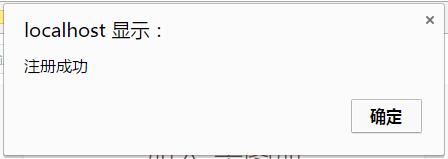


图5.2 正确填写表单结果提示

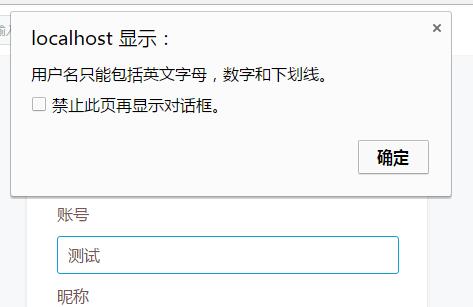


图5.3 账号格式错误提示

1. 登录

如图5.4所示，填写错误则会提示“密码或账号错误”，反之跳转到首页，如图5.5所示，右上角会出现用户的昵称表示身份。



图5.4 密码或者账户错误提示



图5.5 登录成功跳转首页

1. 修改信息

如图5.6所示，填写成功会出现“修改成功”的提示，若未按要求进行输入，则会出现如图5.7所示的“修改失败”提示。



图5.6 修改个人信息测试（正确）



图5.7 修改个人信息测试（错误）

1. 图册管理测试
2. 新建图册

若设置成功，则会出现如图5.8所示新建的图册在页面。若填写信息不完整会出现相应提示，如图5.9所示。

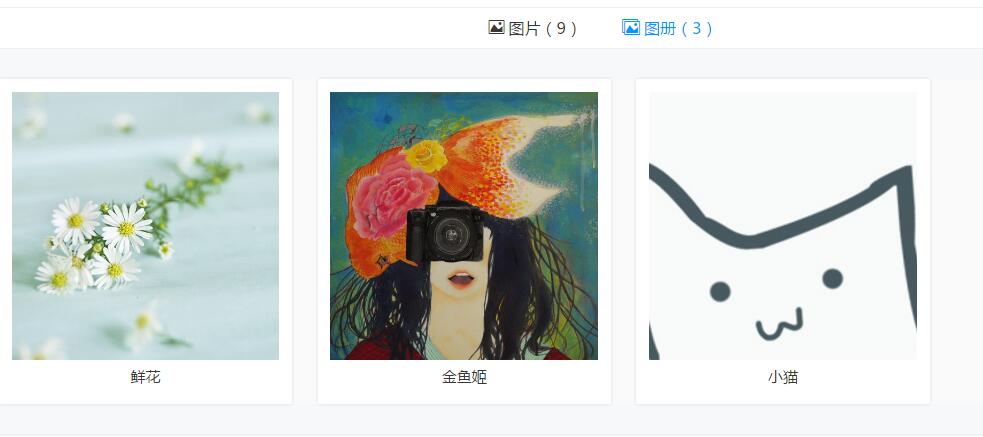


图5.8 新建图册测试（正确）

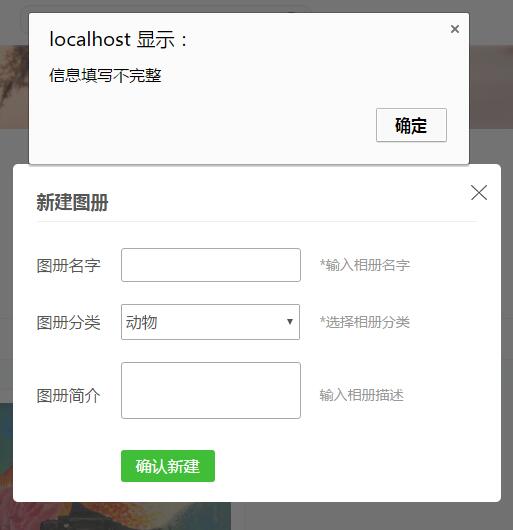


图5.9 新建图册测试（错误）

1. 修改图册信息

若设置成功，则会出现如图5.10所示修改的图册成功的提示。



图5.10 修改图册测试（正确）

1. 删除图册

若删除成功，则会出现如图5.11所示删除成功的提示。若图册还存在图片，则不允许直接删除，如图5.12所示。



图5.11 删除图册测试（正确）



图5.12 删除图册测试（错误）

1. 设置图册封面

设置成功则会出现如图5.13的提示和封面的变化。设置失败则会提示设置失败的提示（一般是数据库连接或网络错误造成）。



图5.13 设置封面测试（错误）

1. 图片管理测试
2. 上传图片

上传成功则会出现如图5.14的提示。上传信息不符合则会提示会出现如图5.15的提示上传出错的提示。

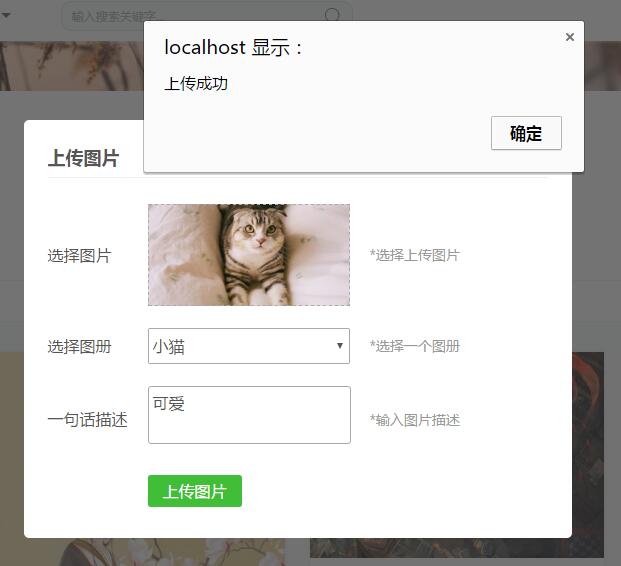


图5.14 上传图片测试（正确）

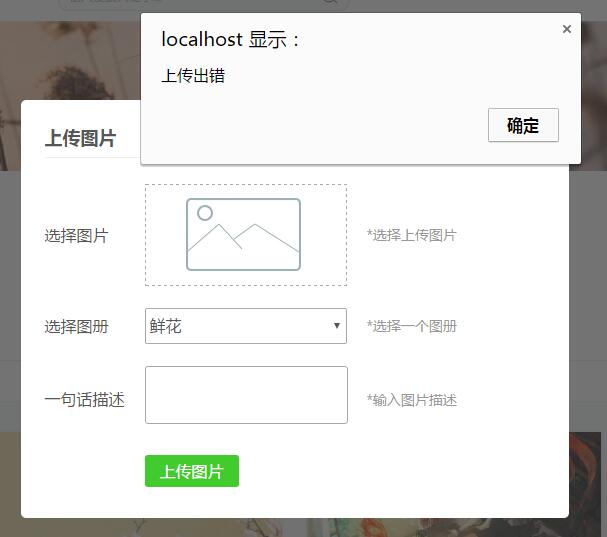


图5.15 上传图片测试（错误）

1. 修改图片信息

修改成功则会出现如图5.16的提示。修改信息不符合则会提示会出现修改出错的提示。



图5.16 修改图片测试（正确）

1. 采集图片

采集成功则会出现如图5.17的提示，且主页能看到自己的如图5.18的采集信息。

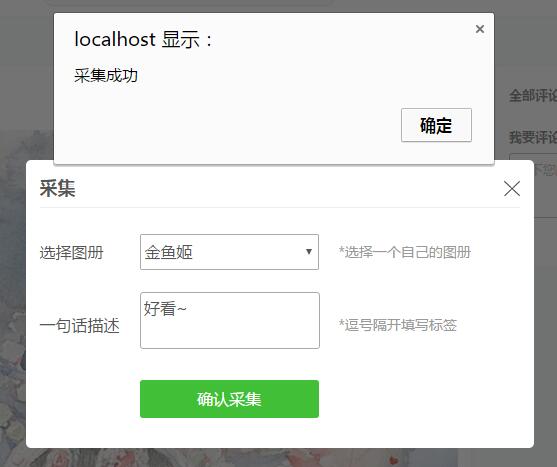


图5.17 采集图片测试（正确）



图5.18 采集图片显示

1. 喜欢图片

喜欢图片，则按钮样式会发生如图5.19的变化，页脚的喜欢数也发生变化。再次点击可以取消喜欢，恢复原来的样子。



图5.19 喜欢图片测试

1. 评论图片

评论图片成功则会提示如图5.20，并如图5.21刷新页面显示评论。



图5.20 评论图片测试



图5.21 评论图片测试

1. 删除图片

评论图片成功提示如图5.22的提示，并跳转到个人主页页面。



图5.22 删除图片测试

1. 搜索功能测试
2. 图册搜索

搜索图册“花”结果如图5.23的提示。

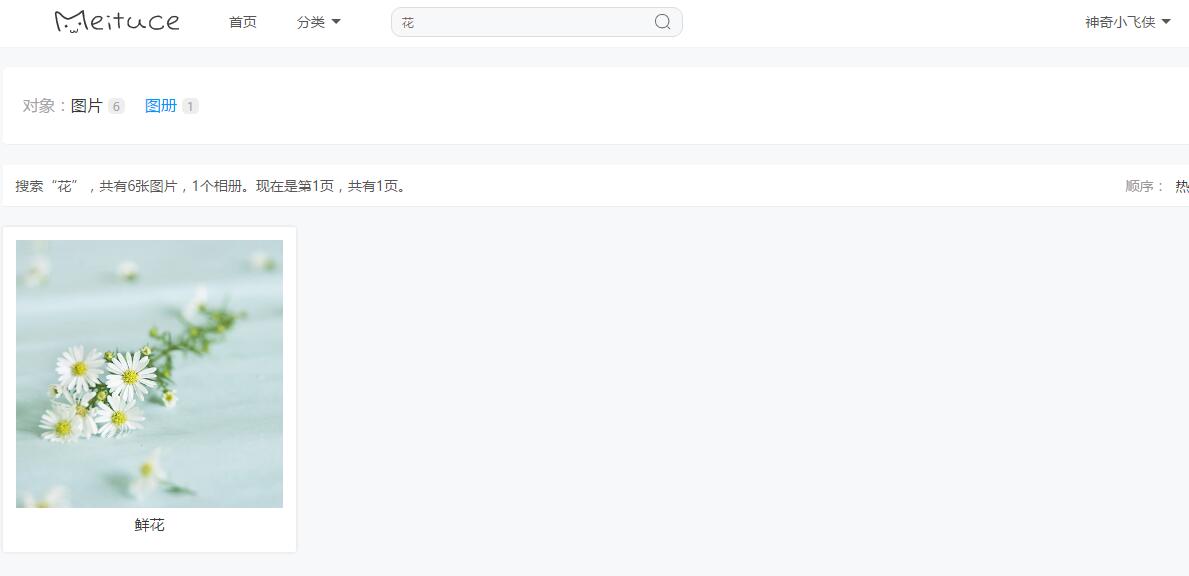


图5.23 搜索“花”相关图册测试

1. 图片搜索

搜索图片“花”结果如图5.24的提示。

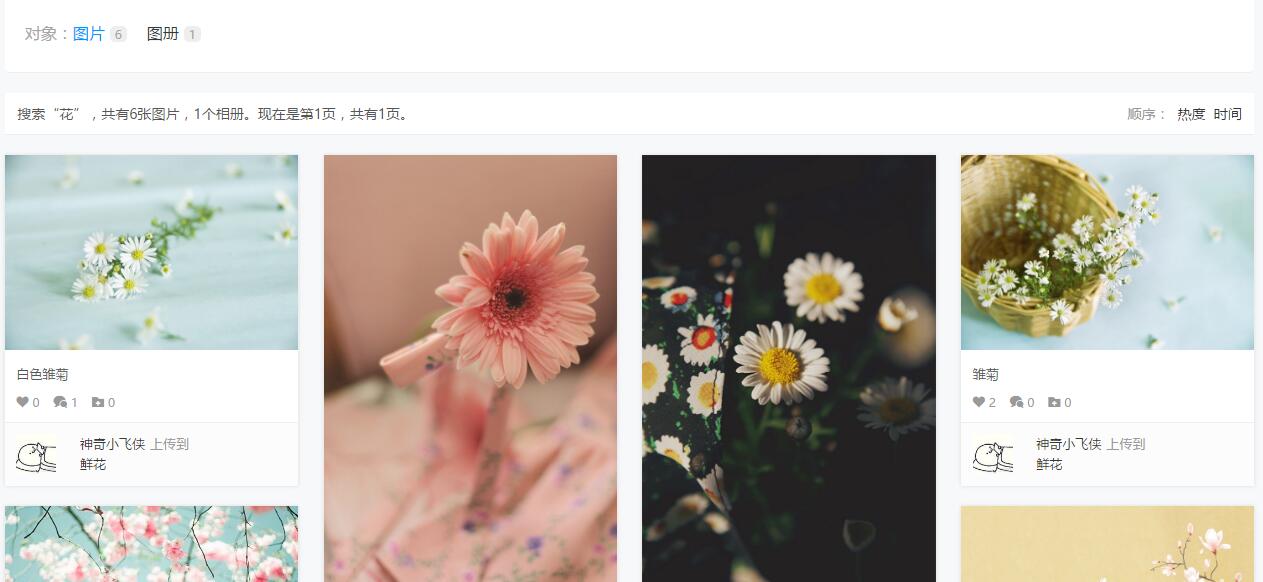


图5.24搜索“花”相关图片测试

6 结束语

本网站基本实现了预期的效果和功能，不过由于时间的紧凑，还有一些细节处理的不是很到位。我通过此次实训项目的制作，感触颇多：

1. 实现一个系统的过程中，需求分析是核心，它决定了我们系统的基本功能和结构。要想做出一个实用、能解决问题的系统，就不能忽视对需求的分析。对于我们不熟悉的领域，我们更应该多去查阅资料，去接近相关行业的人员，进行调查和总结。若条件不太允许，我们也可以去体验使用一些类似的、现成的相关系统，从而帮助我们设计的更加合理和实用。
2. 这段实训期间自己一个人完成了一个项目，我从开始对PHP只是了解皮毛到完成后感觉自己对PHP的领悟和熟练度都提高了很多。我们边学边做，在学中实践，又在实践中体会知识的运用。还把这两年学习的专业课程，软件工程、数据库、数据结构等等里面的知识都综合运用了起来，更让人印象深刻。
3. 在整个项目的设计过程中，我期望它能完成“更多”的功能，但我一味想着实现这样的功能，却总是忽略细微的地方而造成功能不能很好的实现，缺乏思维的缜密。包括数据库的设计上，我也是一波三折，改了很多次。我在这段时间内，也浏览了很多其他同学的设计制作，并吸取他们优秀的地方。总之，学习是一个长久的过程，想要变得更好还需继续努力。

参考文献

[1]单东林，张晓非，魏然.锋利的JQuery[M].北京：人民邮电出版社，2014

[2]王珊，萨师煊.数据库系统概论.第5版[M].北京：高等教育出版社，2014.

[3]唐汉明，翟振兴，关宝军，王洪权.深入浅出MySQL:数据库开发、优化与管理维护.第2版 [M]北京：人民邮电出版社，2014

[4]Luke Welling，Laura Thomson.PHP和MySQL WEB开发.第四版[M].北京： 机械工业出版社 2015.

致谢

在本次的软件系统设计基础实训中，感谢姚文老师对我们的帮助和辅导。姚文老师细心负责也很热心，能及时帮我们回答各种疑问。由于本次基础实训是以数据库为核心而开设的，在此感谢授于我们数据库设计课程的陈曦老师，让我们脑海中存有一个扎实的数据库理论体系。也很感谢带我入门PHP的湛新霞老师，还有在课堂中给予我们许多软件设计思想的胡立辉老师。最后再特别感谢我的指导老师邓广慧，指引我们不走弯路，让我们顺序地完成了此次实训。

# 附录（部分程序代码）

实现图片上传预览的JS代码：

function photoPreview(file)

{

var prevDiv = document.getElementById('file-btn');

if (!/.(gif|jpg|jpeg|png|GIF|JPG|png|BMP|bmp)$/.test(file.value))

{

return false;

}

if (file.files && file.files[0]) {

var reader = new FileReader();

reader.onload = function(evt) {

prevDiv.style = "background-image:url(" + evt.target.result + ")";

}

reader.readAsDataURL(file.files[0]);

} else {

prevDiv.style = "background-image:url(" + file.value + ")";

}

}

部分函数设计代码：

<?php

function go\_to\_page($url,$text)

{

if(!empty($text))

{

echo "<script>alert('".$text."');</script>";

}

echo "<script>window.location.href='".$url."';</script>";

}

function go\_to\_404()

{

header("Location: ./404.php");

exit;

}

function is\_login($username)

{

if(!isset($username))

{

go\_to\_page("./login.php");

exit;

}

}

//首页分类展示

function show\_header\_category($conn)

{

$ca\_result = mysqli\_query($conn,"select \* from category");

while($row = mysqli\_fetch\_assoc($ca\_result))

{

echo "<li><a href='list.php?category=".$row['ca\_id']."'>".$row['ca\_name']."</a></li>";

}

}

//头部操作按钮显示

function show\_header\_btn($username,$nickname)

{

if($username)

{

echo "<a class='user-menu'>".$nickname."<span class='caret'></span><ul class='sub-nav'><li><a href='user.php?id=".$username."'>个人主页</a></li><li><a href='./self\_set.php'>账号设置</a></li><li><a href='loginout.php'>退出登录</a></li></ul></a>";

}

else

{

echo "<a href='./login.php' class='login'>登录</a><a href='./register.php' class='register'>注册</a>";

}

}

//显示select的图片分类??

function show\_select\_category($conn)

{

$query = "select \* from category";

$ca\_result = mysqli\_query($conn,$query);

while($row = mysqli\_fetch\_assoc($ca\_result))

{

if($photo\_info['p\_category'] == $row['ca\_id'])

echo "<option value='".$row['ca\_id']."' selected='selected'>".$row['ca\_name']."</option>";

else

echo "<option value='".$row['ca\_id']."'>".$row['ca\_name']."</option>";

}

}

//显示编辑图片分类

function show\_edit\_catogory($conn)

{

$query = "select \* from category";

$ca\_result = mysqli\_query($conn,$query);

while($row = mysqli\_fetch\_assoc($ca\_result))

{

if($row == $album\_result['a\_category\_id'])

echo "<option value='".$row['ca\_id']."' selected='selected'>".$row['ca\_name']."</option>";

else

echo "<option value='".$row['ca\_id']."'>".$row['ca\_name']."</option>";

}

}

//查询该分类是否存在

function check\_category\_exist($conn,$category)

{

$category\_result = mysqli\_fetch\_assoc(mysqli\_query($conn,"select \* from category where ca\_id=".$category));

if(isset($category\_result))

{

return true;

}

return false;

}

//输出图片列表

function display\_photo\_box($conn,$result)

{

while($row = mysqli\_fetch\_assoc($result))

{

if(!get\_user\_status($conn,$row['p\_user\_id']))

continue;

$p\_username = get\_user\_info($conn,$row['p\_user\_id']);

$p\_album = get\_album\_info($conn,$row['p\_album\_id']);

$p\_comment = get\_comment\_num($conn,$row['p\_id']);

$like\_num = get\_like\_num($conn,$row['p\_id']);

$pick\_num = get\_pick\_num($conn,$row['p\_id']);

echo "<div class='photo-box'>";

echo "<a class='photo' href='./photo.php?id=".$row['p\_id']."'><img src='photo/".$row['p\_url']."'/></a>";

echo "<div class='photo-info'><p>".$row['p\_des']."</p>";

echo "<div class='photo-info-detail'><p><span class='icon-heart'></span>".$like\_num."</p><p><span class='icon-bubbles'></span>".$p\_comment['num']."</p>";

if(!isset($row['p\_from\_photo']))

echo "<p><span class='icon-folder-plus'></span>".$pick\_num['num']."</p>";

echo "</div><div class='clear'></div></div><div class='user-info'>";

echo "<a class='user-image' href='./user.php?id=".$p\_username['u\_id']."'><img src='";

if(empty($p\_username['u\_head\_image'])) echo "images/default.jpg'";

else echo "photo/user\_avatar/".$p\_username['u\_id'].".jpg";

echo "'/></a><div class='user-detail'><a href='user?id=".$row['p\_user\_id']."'>".$p\_username['u\_nickname'];

if(!isset($row['p\_from\_photo']))

echo "</a>上传到</br>";

else

echo "</a>采集到</br>";

echo "<a href='album.php?id=".$row['p\_album\_id']."'>".$p\_album['a\_name']."</a>";

echo "</div><div class='clear'></div></div></div>";

}

}

//用户ID -> 用户ID，昵称和头像

function get\_user\_info($conn,$id)

{

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select u\_id,u\_nickname,u\_head\_image from user where u\_id='".$id."'"));

return $info;

}

//用户ID -> 用户ID，昵称和头像

function get\_user\_status($conn,$id)

{

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select u\_status from user where u\_id='".$id."'"));

return $info['u\_status'];

}

//图册ID -> 图册名称

function get\_album\_info($conn,$id)

{

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select a\_name from album where a\_id=".$id));

return $info;

}

//图片ID -> 图片评论数量

function get\_comment\_num($conn,$id)

{

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select count(\*) as num from comment where p\_id =".$id));

return $info;

}

//图片ID -> 返回喜欢数量

function get\_like\_num($conn,$id)

{

$num=0;

//看是否有被采集，是则计算采集后被点赞的总数

$info = mysqli\_query($conn,"select p\_id from photo where p\_from\_photo=".$id);

while($row = mysqli\_fetch\_assoc($info))

{

$ex = mysqli\_fetch\_assoc(mysqli\_query($conn,"select count(\*) as num from add\_like where p\_id = ".$row['p\_id']));

$num+=$ex['num'];

}

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select count(\*) as num from add\_like where l\_photo\_id = ".$id));

$num+=$info['num'];

return $num;

}

//图片ID -> 返回采集数量

function get\_pick\_num($conn,$id)

{

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select count(\*) as num from photo where p\_from\_photo = ".$id));

return $info;

}

//相册ID -> 相册封面

function get\_album\_cover($conn,$id)

{

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select p\_url from photo where p\_id=(select a\_cover\_photo from album where a\_id = ".$id.")"));

return $info;

}

//类别ID -> 类别名

function get\_ca\_name($conn,$id)

{

$info = mysqli\_fetch\_assoc(mysqli\_query($conn,"select ca\_name from category where ca\_id=".$id));

return $info;

}

?>

图片上传的后台代码：

<?php

include "../include/conn.php";

include "../include/session\_start.php";

include "../include/functions.php";

$photo\_select\_album = $\_POST['photo\_select\_album'];

$photo\_des = $\_POST['photo\_des'];

if(empty($photo\_select\_album) )

{

go\_to\_page("../user.php?id=".$username,"图片信息不全");

exit;

}

$upload\_image = $\_FILES['file-upload'];

$upload\_name = explode('.',$upload\_image['name']);

if(!is\_uploaded\_file($upload\_image['tmp\_name']))

{

go\_to\_page("../user.php?id=".$username,"上传出错");

exit;

}

$query = "select p\_id from photo order by p\_id desc limit 1";

$row = mysqli\_fetch\_assoc(mysqli\_query($conn,$query));

//确定ID号进行取名

if(empty($row))

{

$photo\_url = 1;

}

else

{

$photo\_url = $row['p\_id']+1;

}

$photo\_url = $photo\_url.".".$upload\_name[count($upload\_name)-1];

$upload\_path = "../photo/";

if(move\_uploaded\_file($upload\_image['tmp\_name'],$upload\_path.$photo\_url))

{

$query = "insert into photo(u\_id,a\_id,p\_time,p\_url,p\_des) values('".$username."',".$photo\_select\_album.",'".date("Y-m-d H:i:s")."','".$photo\_url."','".$photo\_des."')";

if(mysqli\_query($conn,$query))

{

go\_to\_page("../user.php?id=".$username,"上传成功");

}

else

{

go\_to\_page("../user.php?id=".$username,"上传出错");

exit;

}

}

else

{

go\_to\_page("../user.php?id=".$username,"上传出错");

exit;

}

?>