

ImageS概要设计报告

|  |  |
| --- | --- |
| 班级： | 计科1604 |
| 小组成员： | 赖思羽 16281165  王晓云 16281174  扎西拉姆 16281179  郑晖 16281183 |
| 日期： | 2019年6月11日 |

目录

[1 引言 4](#_Toc11186884)

[1.1 编写目的 4](#_Toc11186885)

[1.2 参考资料 4](#_Toc11186886)

[2 名词解释 4](#_Toc11186887)

[3 系统总体架构设计 5](#_Toc11186888)

[3.1 数据模块 6](#_Toc11186889)

[3.2 模型层 6](#_Toc11186890)

[3.3 控制器层 6](#_Toc11186891)

[3.4 视图层 7](#_Toc11186892)

[3.5 外部模块 7](#_Toc11186893)

[4 系统软件结构设计 8](#_Toc11186894)

[4.1 总体设计 8](#_Toc11186895)

[4.2 浏览模块 10](#_Toc11186896)

[4.2.1 模块描述 10](#_Toc11186897)

[4.2.2 功能 10](#_Toc11186898)

[4.3 个人账户管理模块 10](#_Toc11186899)

[4.3.1 模块描述 10](#_Toc11186900)

[4.3.2 功能 10](#_Toc11186901)

[4.4 个人主页管理模块 11](#_Toc11186902)

[4.4.1 模块描述 11](#_Toc11186903)

[4.4.2 功能 11](#_Toc11186904)

[4.5 社交管理模块 11](#_Toc11186905)

[4.5.1 模块描述 11](#_Toc11186906)

[4.5.2 功能 11](#_Toc11186907)

[5 数据库设计 13](#_Toc11186908)

[5.1 数据模型 13](#_Toc11186909)

[5.1.1 用户类（User） 13](#_Toc11186910)

[5.1.2 小组类（Group） 14](#_Toc11186911)

[5.1.3 小组评论类（GroupComment） 14](#_Toc11186912)

[5.1.4 好友类（Friends） 14](#_Toc11186913)

[5.1.5 专辑类（Album） 14](#_Toc11186914)

[5.1.6 图片类（Picture） 15](#_Toc11186915)

[5.1.7 图片评论类（PicComment） 15](#_Toc11186916)

[5.1.8 图片标签类（PicTag） 15](#_Toc11186917)

[5.2 数据字典 15](#_Toc11186918)

[5.3 ER图 20](#_Toc11186919)

[6 接口设计 21](#_Toc11186920)

[6.1 用户接口 21](#_Toc11186921)

[6.2 内部接口 22](#_Toc11186922)

[6.3 软件接口 22](#_Toc11186923)

[6.4 硬件接口 23](#_Toc11186924)

[7 实现与测试计划 23](#_Toc11186925)

[8 版本历史 23](#_Toc11186926)

# 引言

## 编写目的

本文档旨在将软件系统功能进一步细化，将原本复杂的功能拆分成不同的模块，并逐步开发健壮的系统结构，使软件系统需求能够很好的转化为未来设计，并进一步提高软件各方面性能。本文档依据高内聚低耦合的原则，将结构分解为不同的模块，各模块分别负责自己的功能，且模块与模块之间，模块与环境之间的调用关系尽可能简单。同时，在进行概要设计分析的过程中，发现并解决了系统中各个模块之间功能重合、冗余和功能短缺的部分，为详细设计和代码编写提供了保障。

本文档的目标读者为本软件的开发人员和其他相关人员。

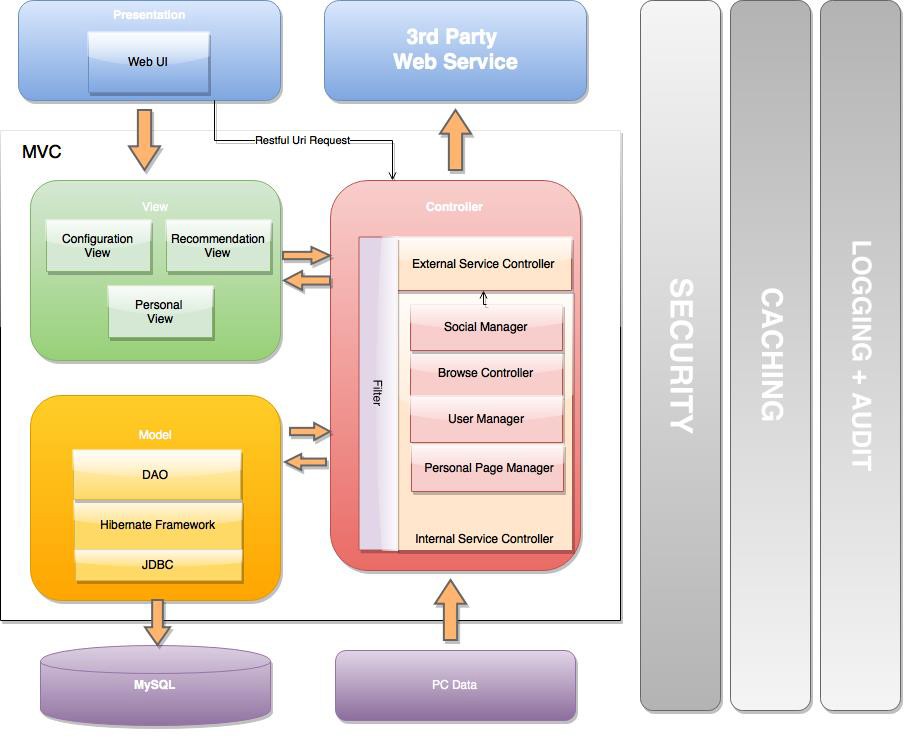
## 参考资料

1. 《软件工程导论（第六版）》，张海藩、牟永敏编著，清华大学出版社。
2. 《8组ImageS 项目需求分析报告v0.3》［2019-05-12］
3. 《计算机软件产品开发文件编制指南（GB8567-88）》

# 名词解释

|  |  |
| --- | --- |
| 词汇 | 说明 |
| 图片 | 网站中所有的经审核可以被用户查看的图片，也包括用户本地想要上传到网站图片。 |
| 专辑 | 网站中相同或者相近主题图片的集合。专辑有说明信息，专辑中的图片也可以随时被该专辑的创建者增加或者删除。 |
| 小组 | 不同的用户在小组中交流或者分享各自喜欢的图片。每个小组都有不同的主题，小组的创建者有权提出某个小组成员或者解散小组。 |
| 个人主页 | 已登陆用户可以有自己的个人主页，个人主页当中包括自己收藏的图片，建立的专辑，发起的和加入的小组。 |
| 好友 | 一个用户可以在网站中点击另一用户头像旁边的“♥” 标志关注对方用户。 |
| 未登录用户 | 未注册用户或者已注册但是没有登录的用户，只可以在网站中浏览查询图片，不具有其他的权利。 |
| 已登陆用户 | 在该网站中已经注册并且已经登录的用户，除浏览和查询图片外，享有建立个人主页、建立专辑、建 立（发起）小组、上传图片等一系列权利。 |
| 第三方社交平台 | 系统设定为腾讯 QQ 和微信平台。用户可以把自己 喜欢的图片通过第三方平台的 API 分享到上述平台 的个人空间和朋友圈。 |

# 系统总体架构设计



系统总体架构图

以需求分析说明书的 2.2 节为依据，进一步详细设计系统的总体架构设计。

本系统服务器端设计以实现各模块间的低耦合及模块内部的高内聚为导向，采用 MVC 作为总体架构；客户端向服务端的请求采用 Restful 设计风格，力求访问的灵活性；服务器端访问 MySQL 数据库进行数据交互；同时在系统层面设计了安全保障模块、缓存模块、日志和验证模块，为系统平台的安全和高效保驾护航。

## 数据模块

本系统作为基于 Web 的图片分享平台，对于大量的高清图片数据、专辑信息、小组信息、用户个人信息、用户访问记录等数据的大量存储和快速访问有着较高的要求。本系统使用 MySQL 5.6 关系型数据库存储信息，通过 JDBC 完成对数据库的访问。MySQL 数据库作为一个快速、多线程编程的数据库，不仅满足本系统对于访问速度的要求，而且具有强大的查询功能，可以在同一查询中混用来自不同数据库的表，查询方便快捷。本系统支持用户上传本地图片，支持 PNG, BMP, JPEG, JPG 等常见图片格式，上传之后存入 MySQL 数据库。

## 模型层

模型层负责处理本系统中应用程序的数据逻辑部分，对数据对象进行存取操作。本模块通过JDBC访问MySQL数据库，采用Hibernate框架自动完成数据库表到Java类的映射。为了构建一个健壮的J2EE应用，本模块采用DAO作为接口，统一管理上层逻辑对于数据源的访问操作，使得上层逻辑无需关心数据访问的细节，同时扩展了系统的可维护性、可拓展性。

## 控制器层

控制器层是本系统处理用户请求的核心层，负责对用户的http请求作出响应，处理本系统的核心业务逻辑，转发用户请求。控制器层通过Filter过滤来自用户的请求，检查用户请求的数据格式及用户当前的登录状态，并根据判断结果对请求进行转发。用户请求被转发到内部服务、外部服务两大模块，外部服务负责获取与第三方社交平台的接口，并将用户图片分享到第三方社交平台；内部服务分为社交管理、浏览管理、个人账户管理、个人主页管理四大部分，是本系统的核心功能逻辑。

* 社交管理：负责处理收藏、评论、分享、关注好友、加入小组等社交业务。
* 浏览管理：负责获取用户个人和系统推荐的图片、专辑、小组数据。
* 个人账户管理：负责登录、登出以及个人信息的配置管理业务。
* 个人主页管理：负责个人图片、专辑和小组的配置管理业务。

## 视图层

视图层是本系统的数据显示部分，是与用户交互的接口。负责系统的前端业务逻辑，支持CSS，JavaScript，JSP文件，将用户请求转发给Controller层，并显示经过Controller处理的业务。

## 外部模块

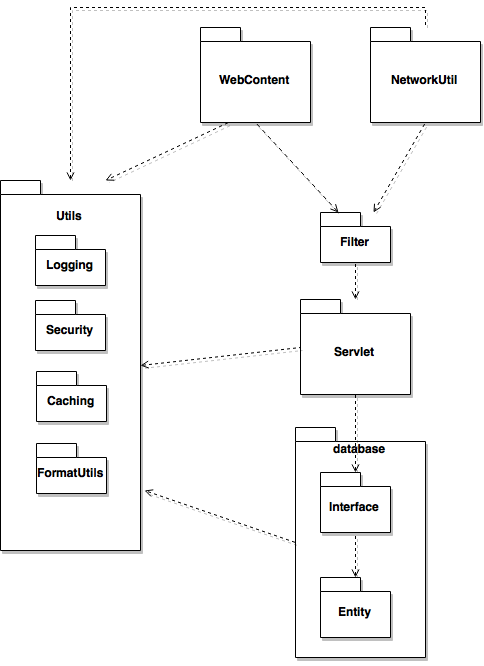
本系统设计已安全性作为基本准则，同时力求为用户提供优化的用户体验，在系统层面设置了安全保障模块、缓存模块和日志模块。安全模块（即 WAF），是作用在 web 服务器上的访 问安全措施，是一种辅助性防御的手段，对于网络应用的安全至关重要，负责拦截黑客入侵行为，如 SQL 注入、跨站脚本、路径操纵、上传/利用网页木马等，并将拦截日志上报到日志模块。

日志模块是系统的必备模块，全面到位的 log 信息决定了系统的健壮性，本系统采用 Apache log4j 作为日志记录工具，对系统的状态变化、业务异常、入侵拦截等进行记录，有助于高效定位问题，便于系统的维护和改进。

缓存模块优化了本系统高清图片的加载速度，本应用为 html 5 应用，采用 Application Apache 进行本地静态缓存，对高清图片加载速度进行优化。

# 系统软件结构设计

## 总体设计



系统包图

根据3. 中系统总体架构的详细设计进行系统包图的设计，总体分为六大模块：

1. **database Package：**

是本系统的数据业务维护模块，对应于系统总体架构中的Model 层。本系统采用 Hibernate ORM 框架，完成数据库表格到 Entity 类的映射，生成 Entity 包，其中的 Entity 类 通过 JDBC 对数据库表进行操作。为了降低系统各模块之间的耦合度，采用 DAO 的设计模式，在 Entity 之上抽象出 Interface 接口包，负责响应上层控制逻辑中对于数据库的操作，对用户请求进行处理，并访问基础 Entity 类。

1. **Servlet Package：**

是本系统的核心业务逻辑包，对应于系统总体架构中的 Controller 层。

Servlet 包负责五大功能模块：

外部服务：负责获取第三方平台 API，并将图片分享到第三方平台，并转发给 NetworkUtil 包；

社交管理：负责处理收藏、评论、分享、关注好友、加入小组等社交业务。 浏览服务：负责获取系统为用户推荐的图片、专辑、小组，并转发给 WebContent 包。

用户服务：负责用户的登录、登出、注册、个人信息修改。

个人主页管理服务：负责个人图片、专辑和小组的配置管理业务。

1. **Filter Package：**

对用户请求作出统一检查，对应于系统总体架构中的 Controller 层。Filter 包负责对用 户请求的数据格式进行检查；负责对用户当前的登录状态进行检查，并根据不同的用户登录状 态决定转发页面。

1. **WebContent Package：**

对应于系统总体架构的 View 层，包括系统的 CSS、图片等静态资源及 JavaScript，JSP 等 动态资源等，直接与用户的进行交互。

1. **NetworkUtil Package：**

负责获取第三方社交平台的接口和认证。

1. **Utils Package：**

对应于系统总体架构中的外部模块，包括系统的通用功能。

## 浏览模块

### 模块描述

为用户提供系统推荐查看个人主页,并且查看网站反馈的图片资源。

### 功能

（1）推荐最热图片；

（2）推荐最新图片；

（3）按照用户偏好推荐图片；

（4）推荐最热专辑；

（5）推荐最新专辑；

（6）按照用户偏好推荐专辑；

（7）推荐最热小组；

（8）按照用户偏好推荐小组；

（9）查看图片详情；

（10）查看专辑详情；

（11）查看小组详情；

## 个人账户管理模块

### 模块描述

负责用户注册、登录、登出，以及修改个人信息。

### 功能

（1）登录

（2）登出

（3）注册

（4）修改个人信息

## 个人主页管理模块

### 模块描述

用户可以在个人主页上传图片，并以专辑的形式归类图片，同时以管理员的身份管理讨论小组。

### 功能

（1）用户可以发起小组，并且修改小组的名称和主题

（2）用户可以删除自己发起的小组

（3）用户可以修改专辑的名称、描述和封面图片

（4）用户可以删除专辑

（5）用户可以新建专辑，确定专辑的名称、描述和封面图片

（6）用户可以上传图片，确定图片的名称、标签和描述

（7）用户可以修改图片的名称、标签和描述

（8）用户可以删除图片

## 社交管理模块

### 模块描述

为用户提供收藏、评论、分享图片、关注好友动态并且针对感兴趣的话题发起讨论小组等社交功能。同时负责获取第三方社交平台 API，支持分享图片到第三方平台。

### 功能

（1）用户可以在图片详情页面发表对该图片的评论。

（2）用户可以在专辑详情页面发表对该专辑的评论。

（3）用户可以在小组详情页面发表对该小组的评论。

（4）用户可以收藏喜爱的图片。

（5）用户可以收藏喜爱的专辑。

（6）用户可以加入感兴趣的小组。

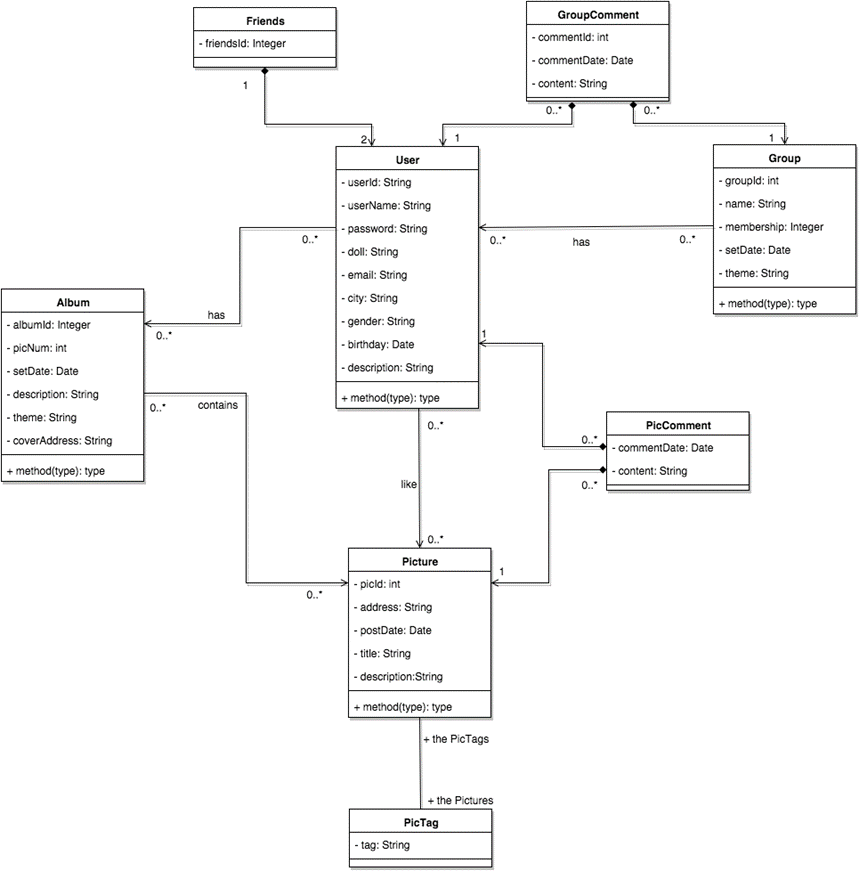
（7）用户可以关注感兴趣的用户。

（8）用户可以将图片及说明分享到第三方社交平台。

（9）用户可以发起小组。

# 数据库设计

## 数据模型



### 用户类（User）

编号：CD1

职责：存储和管理用户基本信息。

属性：用户 ID，昵称，密码，头像，邮箱，所在城市，性别，生日，个人描述。 说明：用户类和图片类单向关联，一个用户可以收藏多张图片。用户类的所有属性都是持久的，它存储了注册用户的基本信息。

### 小组类（Group）

编号：CD2

职责：存储和管理小组基本信息、小组成员。

属性：小组 ID，名称，创建日期，主题。

说明：小组类和用户类存在聚合关系，一个小组由多个用户组成。 小组类的所有属性都是持久的，它存储了小组及成员的信息。

### 小组评论类（GroupComment）

编号：CD3

职责：存储用户在小组中的评论发言。

属性：评论 ID，评论日期，评论内容。

说明：小组评论类与用户存在关联关系，一个用户可以发表多条小组评论；小组评论类与小组存在关联关系，一个小组可以被多次评论。

### 好友类（Friends）

编号：CD4

职责：存储用户之间的好友关系。

属性：好友 ID，用户 1ID，用户 2ID。

说明：好友类与用户类存在聚合关系，一个好友由两个用户组成，在类的实例化过程中，两个类所扮演的角色不同。

### 专辑类（Album）

编号：CD5

职责：存储专辑的基本信息。

属性：专辑 ID，图片数量，创建日期，描述，主题。

说明：用户类和专辑类存在关联关系，一个用户可以收藏或创建多个专辑。

### 图片类（Picture）

编号：CD6

职责：存储图片基本信息。

属性：图片 ID，地址，上传日期，标题，描述。

说明：图片类中的所有属性都是持久化的，它负责存储图片的基本信息。

### 图片评论类（PicComment）

编号：CD7

职责：存储用户对于某张图片的评论。

属性：评论日期，内容。

说明：图片评论类和用户存在关联关系，一个用户可以发起多条评论。图片评论类和图片之间存在关联关系，一个图片可以被多次评论。

### 图片标签类（PicTag）

编号：CD8

职责：存储图片的标签。

属性：标签内容。

说明：标签类和图片之间存在关联关系，一个图片有多个标签，一个标签标记多个图片。

## 数据字典

Album（专辑）

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 描述 |
| albumId | Integer | 专辑 ID，主键 |
| user | User | 新建专辑的用户 |
| picNum | int | 专辑拥有的图片数目 |
| setDate | Date | 专辑的创建日期 |
| description | String | 专辑的描述 |
| theme | String | 专辑主题 |
| coverPath | String | 封面图片的本地路径 |

AlbumIncluded（专辑－图片）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| albumId | int | 专辑，主键 1 |
| picId | Integer | 专辑包含的图片，主键 2 |

AlbumLiked（专辑－用户关系集：收藏该专辑的用户）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| albumId | int | 专辑，主键 1 |
| userId | Integer | 收藏专辑的用户，主键 2 |

Friends（用户－用户）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| friendId | Integer | 好友用户，主键 1 |
| myId | int | 用户本人，主键 2 |

GroupComment （小组－评论）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| commentId | Integer | 评论 ID，主键 |
| group | Group | 被评论的小组 |
| user | User | 发起评论的用户 |
| commentDate | Date | 评论日期 |
| content | String | 评论内容 |

Group（小组）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| groupId | Integer | 小组 ID，主键 |
| user | User | 发起小组的用户 |
| name | String | 小组名称 |
| membership | Integer | 小组成员数 |
| setDate | Date | 发起小组的日期 |
| theme | String | 小组的主题 |

GroupMem （小组－用户关系集：小组及其成员）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| groupId | Integer | 小组，主键 1 |
| userId | Integer | 小组包含的用户，主键 2 |

User（用户）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| userId | Integer | 用户 ID |
| userName | String | 用户姓名 |
| password | String | 用户密码 |
| avatar | String | 用户头像 |
| email | String | 用户邮箱地址 |
| city | String | 用户所在城市 |
| gender | String | 用户性别 |
| birthday | Date | 用户生日 |
| discription | String | 用户个人描述 |

Pictures（图片）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| picId | Integer | 图片 ID，主键 |
| user | User | 上传图片的用户 |
| path | String | 图片的本地地址 |
| uploadDate | Date | 上传日期 |
| title | String | 图片标题 |
| description | String | 图片描述 |

PicComment （图片评论）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| commentId | Integer | 关系集 ID，主键 |
| pictures | Pictures | 被评论的图片 |
| user | User | 发起评论的用户 |
| commentDate | Date | 评论日期 |
| content | String | 评论内容 |

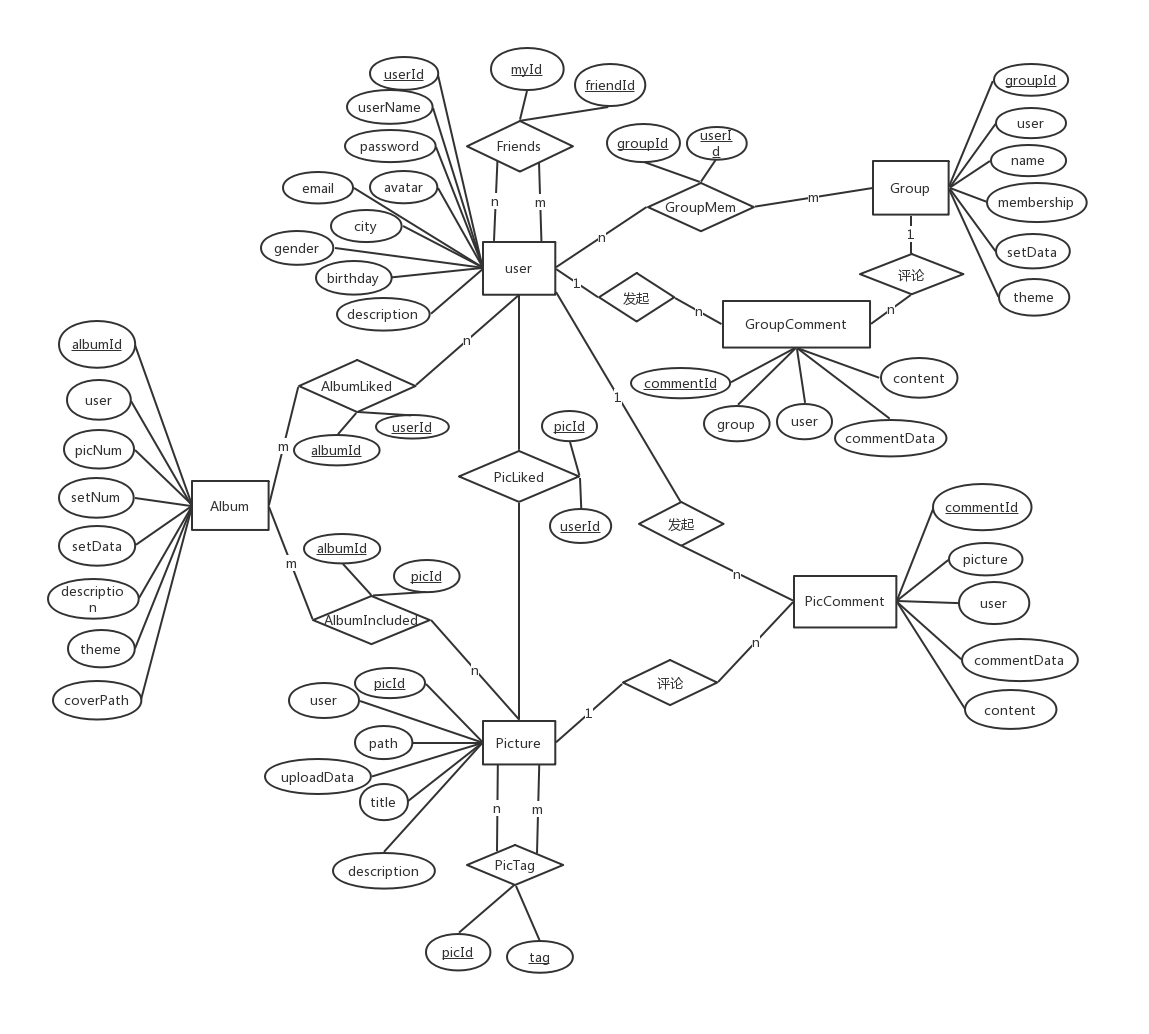
PicLiked （图片－用户关系集：图片与收藏它的用户）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| picId | Integer | 被收藏的图片，主键 1 |
| userId | Integer | 收藏图片的用户，主键 2 |

PicTag （图片－标签关系集：图片所拥有的标签）

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| tag | String | 图片对应的标签，主键 1 |
| pictureId | Picture | 图片，主键 2 |

## ER图



# 接口设计

## 用户接口

|  |  |
| --- | --- |
| 社交管理模块 | 评论图片接口 |
| 评价小组接口 |
| 评价专辑接口 |
| 收藏图片接口 |
| 收藏专辑接口 |
| 关注好友接口 |
| 加入小组接口 |
| 分享到第三方平台接口 |
| 浏览模块 | 最热图片推荐接口 |
| 最热小组推荐接口 |
| 最热专辑推荐接口 |
| 最近图片推荐接口 |
| 最近小组推荐接口 |
| 最近专辑推荐接口 |
| 按照喜好图片推荐接口 |
| 按照喜好小组推荐接口 |
| 按照喜好专辑推荐接口 |
| 查看图片详情 |
| 查看小组详情 |
| 查看专辑详情 |
| 个人主页查看上传图片接口 |
| 个人主页查看收藏图片接口 |
| 个人主页查看新建专辑接口 |
| 个人主页查看收藏专辑接口 |
| 个人主页查看发起小组接口 |
| 个人主页查看用户所在小组接口 |
| 搜索接口 |
| 个人账户管理模块 | 登录接口 |
| 注册接口 |
| 登出接口 |
| 个人信息修改接口 |
| 个人主页管理模块 | 修改小组信息接口 |
| 删除小组接口 |
| 发起小组接口 |
| 删除专辑接口 |
| 新建专辑接口 |
| 修改专辑信息接口 |
| 上传图片接口 |
| 修改图片信息接口 |
| 删除图片接口 |

## 内部接口

|  |  |
| --- | --- |
| 内部接口 | 社交管理模块 |
| 社交管理模块 |
| 个人账户管理模块 |
| 个人主页管理模块 |

## 软件接口

1. 微信公众开发平台接口API

2. 腾讯QQ接口API

3. 新浪微博接口API

## 硬件接口

1. 账户授权协议:OAuth 2.0

2. 网络访问协议:TCP/IP,HTTP

3. 数据压缩:objective-zip

# 实现与测试计划

## 实现计划

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **任务名称** | **主要负责人** | **开始时间** | **完成时间** | **持续时间** |
| **1** | 文档编辑 | 王晓云 | 2019.5.1 | 2019.6.16 | 47d |
| **2** | 收集录入 | 扎西拉姆 | 2019.6.17 | 20198.6.20 | 3d |
| **3** | 功能模块 | 郑晖 | 2019.6.21 | 2019.6.25 | 4d |
| **4** | 代码编写 | 全体 | 2019.6.26 | 2019.7.15 | 19d |
| **5** | 界面优化 | 赖思羽 | 2019.7.16 | 2019.7.20 | 4d |
| **6** | 软件测试 | 王晓云、扎西拉姆 | 2019.7.21 | 2019.7.31 | 10d |
| **7** | 软件优化 | 郑晖、赖思羽 | 2019.8.1 | 2019.8.10 | 9d |

## 测试计划

### 测试目标

* 测试已实现的产品是否达到设计的要求，包括各个功能点是否已实现，业务流程是否正确。
* 产品规定的操作和运行稳定
* 错误数和缺陷率控制在可接受范围内

### 测试内容

* 数据和数据库完整性测试

调用各个数据库访问方法和进程，并在其中填充有效的和无效的数据（或对数据的请求）。检查数据库，确保数据已按预期的方式填充，并且所有的数据库事件已正常发生；或者检查所返回的数据，确保正当的理由检索到了正确的数据

* 接口测试

确保接口调用的正确性，记录所有软件、硬件接口，记录输入输出数据

* 集成测试

利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：

在使用有效数据时得到预期的结果。

在使用无效数据时显示相应的错误消息或警告消息。

各业务规则都得到了正确的应用。

* 功能测试

利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：

在使用有效数据时得到预期的结果。

在使用无效数据时显示相应的错误消息或警告消息。

各业务规则都得到了正确的应用。

* 用户界面测试

核实以下内容：

通过测试进行的浏览可正确反映业务的功能和需求，这种浏览包括窗口与窗口之间、字段与字段之间的浏览，以及各种访问方法（Tab键、鼠标移动、和快捷键）的使用

窗口的对象和特征（例如，菜单、大小、位置、状态和中心）都符合标准。

* 性能评测

核实所指定的事务或业务功能在以下情况下的性能行为：

正常的预期工作量

预期的最繁重工作量

# 版本历史

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 编写日期 | 说明 | 参与人员 |
| 0.1 | 2019-06-11 | 初稿 | 郑晖 |
| 0.2 | 2019-06-16 | 添加测试计划 | 赖思羽 |
| 0.3 | 2019-06-16 | 修改定稿 | 王晓云、扎西拉姆 |