目录

[一、目标与服务模型 4](#_Toc486760508)

[1.1项目背景 4](#_Toc486760509)

[1.2项目目标 4](#_Toc486760510)

[1.3解决思路 5](#_Toc486760511)

[1.3.1合作方式 5](#_Toc486760512)

[1.3.2分析方法 5](#_Toc486760513)

[1.3.3制定计划 5](#_Toc486760514)

[1.3.4技术手段 6](#_Toc486760515)

[1.4工作范围与交付物说明 6](#_Toc486760516)

[1.4.1工作范围说明 6](#_Toc486760517)

[1.4.2交付物说明 6](#_Toc486760518)

[二、需求分析方案与组织管理 7](#_Toc486760519)

[2.1.1功能性需求分析 7](#_Toc486760520)

[2.1.2非功能性需求分析 27](#_Toc486760521)

[2.2项目管理 29](#_Toc486760522)

[2.2.1项目生命周期与组织 29](#_Toc486760523)

[2.2.2角色与职责 29](#_Toc486760524)

[2.2.3项目评审 29](#_Toc486760525)

[2.2.4任务分解与进度安排 29](#_Toc486760526)

[2.2.5项目过程管理 29](#_Toc486760527)

[2.2.6项目里程碑管理 29](#_Toc486760528)

[2.2.7项目质量管理 29](#_Toc486760529)

[2.2.8项目沟通管理 29](#_Toc486760530)

[2.2.9项目风险管理 29](#_Toc486760531)

[三、技术路线及实现方案 29](#_Toc486760532)

[3.1项目概要设计 29](#_Toc486760533)

[3.1.2系统模块设计 29](#_Toc486760534)

[3.1.3系统接口设计 29](#_Toc486760535)

[3.2数据库设计 38](#_Toc486760536)

[3.3技术实现 42](#_Toc486760537)

[3.4环境搭建 42](#_Toc486760538)

[3.5安全架构 42](#_Toc486760539)

[3.6界面设计 42](#_Toc486760540)

[四、成本模型及可行性分析 43](#_Toc486760541)

[4.1成本模型 43](#_Toc486760542)

[4.2可行性分析 46](#_Toc486760543)

[4.2.1市场分析 46](#_Toc486760544)

[4.2.2政策分析 46](#_Toc486760545)

[4.2.3经济可行性分析 48](#_Toc486760546)

[4.2.4技术可行性分析 48](#_Toc486760547)

[4.2.5操作可行性分析 48](#_Toc486760548)

[4.2.6法律可行性分析 48](#_Toc486760549)

[4.3风险管理 48](#_Toc486760550)

[4.3.1技术风险 48](#_Toc486760551)

[4.3.2组织风险 48](#_Toc486760552)

[4.3.3人员流动风险 48](#_Toc486760553)

[4.3.4控制进度风险 48](#_Toc486760554)

[附录：会议记录摘要 48](#_Toc486760555)

# 一、目标与服务模型

## 1.1项目背景

随着智能手机的高速发展，智能手机经成为大街小巷几乎人手一个的标配。越来越多的人通过手机来处理各种各样的事务。从一开始的互联网兴起到移动互联网的深度发展，求新求变的需求一直都非常旺盛。在图像视觉领域，从素描油画，到拍照录像技术，再到影视直播，图像的技术已经慢慢的渗透到人们工作生活的各个角落。在技术上，随着智能设备的发展，以及摄像头应用领域的不断扩展，人们对图像的认知及应用从浅及深，从单纯的手工修改图片，到自动美图，再到机器学习，每一步的前进都代表着人类对于智能认知的推进。而今各式各样的图像视觉应用，开拓人的视野，也对该领域有了一些更多的期待。

现代人对移动互联网越来越适应，人们也对移动互联网的应用提出了更高的要求。从电脑上的各种作图软件到手机上的各种美图软件，人们能够轻而易举修改自己的图片，并分享到各类社交平台。随着人们对美的要求越来越高，艺术性的美化也渐渐被提了出来。人在欣赏自己所拍的图片之余，也希望能够把自己喜欢的图片做各式各样的风格，如冷色调风格，怀旧风格，冲印照片格，素描风格，油画风格等。而手工制图方进行修改，除了费时费力外，还需要有一的制图技能和美学背景。希望在不掌握专业制图技能的前提下，能够轻易的实现将人随手的图片成希望呈现的风格，这样自己的图片者照片能够以一种别样的风格呈现出来，并分享给好友。

除了创造新的图片的风格之外，人们也希望有更多的有趣的图片美化功能，人们也希望将自己制作的精美图片分享给好友以及广大网友一起欣赏，当看到别人通过较为专业的制图技术美化了一张图片之后，自己也想去尝试人家制作好的图片模板，希望能够一键就能将别的用户的美化参数应用到自己的图片上，从而节省自己无厘头的摸索时间。

## 1.2项目目标

为了满足项目发包方需求，同时为了满足用户对更多有趣新颖的图片美化方式的需求，因此计划开发该图像风格转化系统（图酷APP），一个集滤镜、风格转化、贴纸文字、社交、图库于一体的图像风格转化系统。

1. 战略层目标

开发出较为新颖的图片美化方式，以图像风格转化为主，并提高系统的可维护性与可扩展性，并帮助提升公司的竞争实力。

1. 管理层目标

运用科学的项目团队管理技术手段对项目开发进行管控，降低管理风险，让项目开发有序不乱的进行，同时营造良好的团队文化，让团队成员团结互助。

1. 操作层目标

开发出简单易懂的操作界面，让用户能够上手掌握系统核心功能，在减少用户对图片操作的同时，又能够让图片达到或超出用户的期望。

## 1.3解决思路

### 1.3.1合作方式

项目采用团队合作方式完成。组建一支由一名项目经理、两名后台技术经理、一名前端技术经理，一名用户界面设计经理组成的团队，利用专业的研发环境，实施独立管理的项目团队，所有的任务都围绕一个共同的目标展开，团队成员相互独立，实行任务分解策略，每个成员有自己独自的任务，成员需要将任务精益求精，深入分析项目需求、理解项目目标、评估解决方案，包括技术平台、语言、交付进度等，保证项目的顺利执行。

### 1.3.2分析方法

采用软件工程分析方法和设计方法，把在软件计划期间建立的软件可行性分析求精和细化，分析各种可能的解法，并且分配给各个软件元素。从需求分析，需求提出，需求评审，需求建模，到开发原型系统，通过结构化分析策略，采用“自顶向下，逐步求精”的技术对系统进行划分。

### 1.3.3制定计划

整个图像风格转化系统（图酷APP）开发过程制定如下5个阶段开发计划：

1. 问题的定义及规划

此阶段对软件开发与项目需求共同讨论，主要确定项目的开发目标及其可行性。

1. 需求分析

在确定项目开发可行的情况下，对软件需要实现的各个功能进行详细需求分析。需求分析阶段，将为整个软件项目的开发打下良好的基础。“唯一不变的是变化本身”，同样软件需求也是在软件开发过程中不断变化和深入的，因此，我们必须定制需求变更计划来应付这种变化，以保证整个项目的正常进行。

1. 软件设计

此阶段中要根据需求分析的结果，对整个软件系统进行设计，如系统框架设计、数据库设计等。软件设计一般分为总体设计和详细设计，好的软件设计将为软件程序编写打下良好的基础。

1. 程序编码

此阶段是将软件设计的结果转化为计算机可运行的程序代码。而开发人员必须要制定统一、符合标准的编写规范，提高程序的可读性、易维护性、可扩展性，从而提高程序的运行效率。

1. 系统测试

在软件设计完成之后要进行严密的测试。整个测试阶段分为单元测试、集成测试、系统测试三个阶段进行。测试方法主要有白盒测试和黑盒测试。

### 1.3.4技术手段

本次项目开发计划技术栈如下：

1. 服务端
2. 开发语言：Python
3. 服务器：Nginx
4. 服务器框架：Flask
5. 数据库：MySQL
6. 移动端

这里还差移动端技术栈

## 1.4工作范围与交付物说明

### 1.4.1工作范围说明

本次项目开发工作范围包括：

1. 市场分析
2. 项目可行性分析
3. 需求分析
4. 项目管理
5. 产品与技术研发

### 1.4.2交付物说明

本次开发的交付物包括：

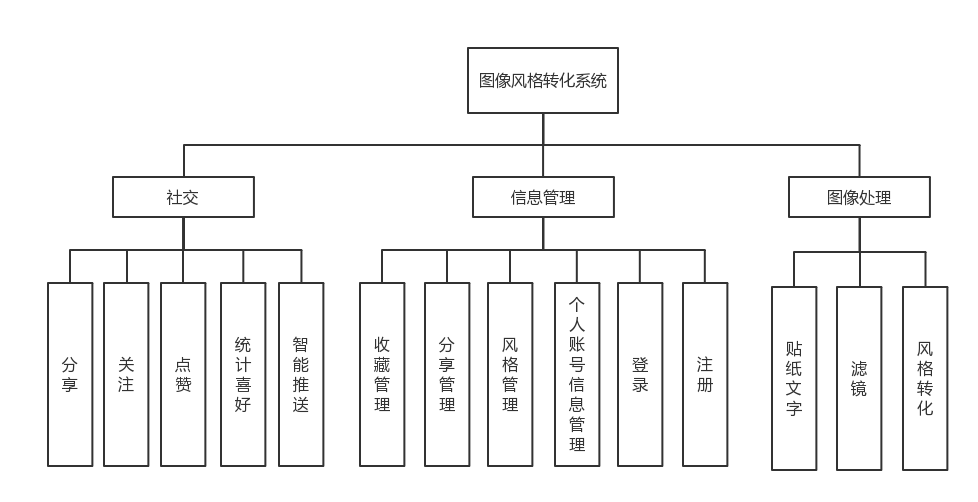
1. 产品原创承诺书。
2. 不超过1500字的方案概要：前言、创意描述、功能简介、特色综述开发工具与技术、应用对象、应用环境、结语。
3. 展示核心内容的PPT：目标与解决思路、问题分析与解决方案、技术路线与技术实现方案、业务模式、人员组织架构、可行性分析。
4. 项目详细解决方案，分为以下四个板块：目标与服务模型、组织管理与业务分析方案、技术路线与实施方案、成本模型及可行性分析。
5. 产品原型：以系统、软件、程序方式提交；以视频的形式演示系统完成情况。
6. 团队完成过程视频：介绍团队完成该项目/任务的过程。

# 二、需求分析方案与组织管理

## 2.1需求分析

### 2.1.1功能性需求分析

#### 2.1.1.1 功能模块图



图像风格转化系统总共分为三大模块，社交模块、信息管理模块、图像处理模块，各功能模块描述如下：

1. 社交模块
2. 分享：用户可以分享经过图像处理过后的图片，或者也可以分享用户创建的风格类型，分享图片的方式有平台内分享或第三方平台（微信、微博、QQ）进行分享，分享风格的方式只有平台内分享。
3. 关注：针对分享图片或图像风格的作者，用户可以关注喜欢的作者。
4. 点赞：针对用户分享的图片或图像风格，用户可以对喜欢的图片或风格进行点赞。
5. 统计喜好：系统会自动统计用户分享风格中用户的点赞数量，获得点赞数量多的风格将会被加入系统内置风格中。
6. 智能推送：根据用户平时的点赞行为和用户的喜好统计，系统将为用户推送更符合用户喜好的作品和信息。
7. 信息管理模块
8. 收藏管理：用户在浏览其他用户的分享内容时，对喜欢的图片可以进行收藏，对于喜欢的风格，用户可以收藏并添加到风格管理中，同时，用户可以通过收藏管理删除已收藏的图片或风格。
9. 分享管理：用户可以通过分享管理查看用户在平台内分享的内容、评论或点赞，同时可以删除分享记录。
10. 风格管理：用户通过风格管理功能查看系统内置的风格、收藏的风格和创建的风格，同时可以删除创建的风格和收藏的风格，但不能删除系统内置风格。用户创建新的风格时，先命名风格名称，然后需要上传一张风格图片和一张内容图片，系统转化风格之后，呈现出三张图片，风格图片、内容图片、转化结果图片，用户可以保存风格，或者选择分享并保存风格。
11. 个人账号信息管理：用户通过个人账号管理可以查看并修改个人账户信息。
12. 登录：用户使用注册账号登录系统。
13. 注册：游客用户注册账号。
14. 图像处理模块
15. 贴纸文字：用户打开摄像头拍摄或从相册中选择一张照片，并且选择需要的贴纸图片或文字，将贴纸图片或文字放置到适当位置，用户可选择保存处理过后的图片，也可分享并保存处理之后的图片。
16. 滤镜：用户打开摄像头拍摄或从相册中选择一张照片，并选择需要的滤镜类型，经过系统处理之后，用户可保存处理后的图片，或者分享并保存处理后的图片。
17. 风格转化：系统内置部分风格类型，用户风格转化时打开摄像头拍摄或从相册中选择一张照片，并选择风格类型，经过系统处理之后，用户可保存处理后的图片，或者分享并保存处理后的图片。

#### 2.1.1.2 UML分析图

##### (1) 用例图

1）总用例图

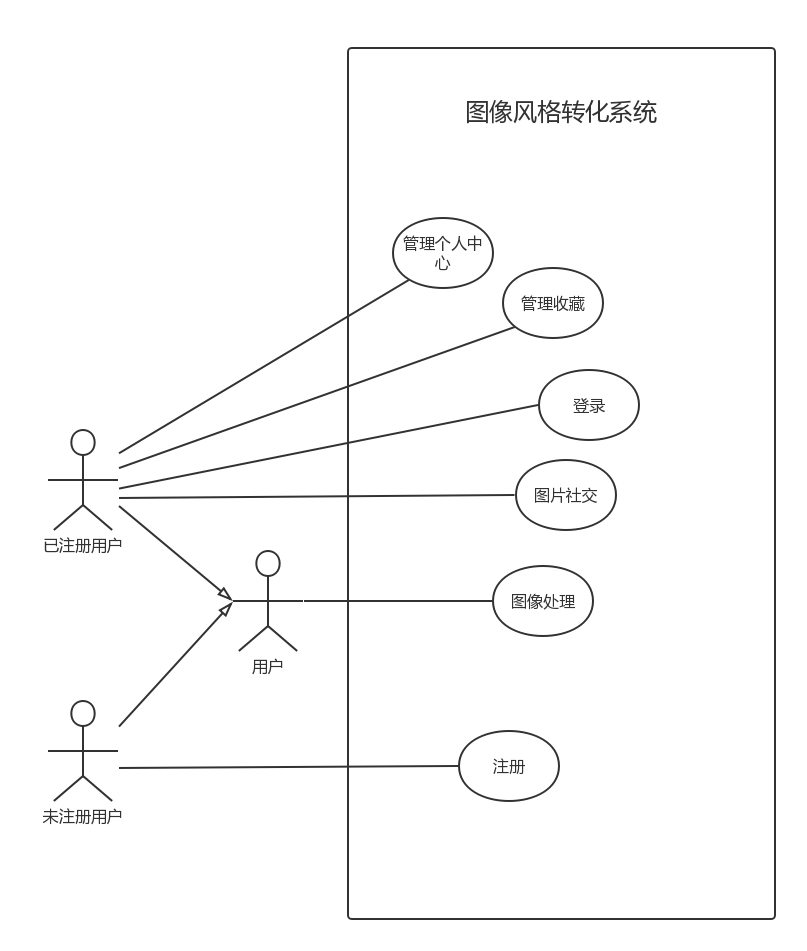


图 2‑0‑1

系统将用户角色划分为已注册用户和未注册用户（游客），其中已注册用户可执行的用例有：管理个人中心、管理收藏、登录、图片社交、图像处理。未注册用户可执行的用例为图像处理、注册。

各用例具体用例及用例描述请看如下：

2）登录注册用例



图 2‑0‑2

|  |  |
| --- | --- |
| **用例名称** | **登录** |
| **用例编号** | 01\_01 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户使用账号密码登录系统 |
| **前置条件** | 用户处于未登录或未注册状态并且需要使用社交功能 |
| **基本事件流** | 1. 用户进入登录界面 2. 选择登录方式 3. 输入账号信息 4. 输入密码 5. 点击确定按钮 6. 登录成功 |
| **扩展事件流** | 2a、选择第三方平台进行登录（QQ、微信、微博）  6a、账号或密码错误，返回事件3 |
| **后置条件** | 登录成功 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **注册** |
| **用例编号** | 01\_02 |
| **参与者** | 未注册用户 |
| **简要说明** | 未注册用户通过注册页面注册账号并登录系统 |
| **前置条件** | 用户处于未登录或未注册状态并且需要使用社交功能 |
| **基本事件流** | 1. 用户进入注册界面 2. 选择注册方式 3. 输入账号信息 4. 输入密码 5. 点击确定按钮 6. 注册成功 |
| **扩展事件流** | 2a、选择第三方平台进行登录（QQ、微信、微博）  3a、账号已存在，返回事件3 |
| **后置条件** | 注册成功，并登录成功 |
| **注释** |  |

3）管理个人中心用例

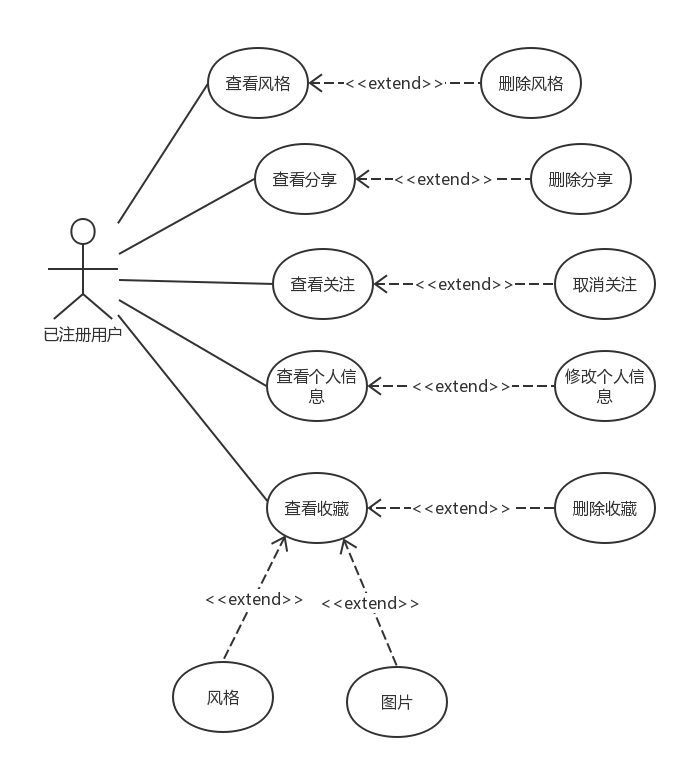


图 2‑0‑3

|  |  |
| --- | --- |
| **用例名称** | **查看风格** |
| **用例编号** | 02\_01 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户在个人中心页面中查看并操作风格 |
| **前置条件** | 用户已正确登录系统 |
| **基本事件流** | 1. 用户进入个人中心页面 2. 点击图片风格类型按钮 3. 查看风格类型 4. 退出风格类型页面 |
| **扩展事件流** | 3a、删除风格类型：  3a1、长按风格类型，弹出菜单  3a2、点击删除  3a3、删除成功  3a2\_1、系统风格无法删除，返回事件2 |
| **后置条件** | 退出风格类型查看页面 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **查看分享** |
| **用例编号** | 02\_02 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户在个人中心页面中查看并操作已分享在该平台的内容 |
| **前置条件** | 用户已正确登录系统 |
| **基本事件流** | 1. 用户进入个人中心页面 2. 点击查看分享按钮 3. 查看分享记录 4. 退出查看分享页面 |
| **扩展事件流** | 3a、删除分享内容：  3a1、长按分享内容，弹出菜单  3a2、点击删除  3a3、删除成功 |
| **后置条件** | 退出分享记录查看页面 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **查看个人信息** |
| **用例编号** | 02\_03 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户在个人中心页面中查看个人信息 |
| **前置条件** | 用户已正确登录系统 |
| **基本事件流** | 1. 用户进入个人中心页面 2. 点击查看个人信息按钮 3. 查看个人信息 4. 退出个人信息页面 |
| **扩展事件流** | 3a、修改个人信息：  3a1、点击编辑信息按钮  3a2、输入需要修改的内容信息  3a3、点击保存按钮  3a4、保存成功 |
| **后置条件** | 退出个人信息页面 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **查看收藏** |
| **用例编号** | 02\_04 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户在收藏页面中查看收藏内容 |
| **前置条件** | 用户已正确登录系统 |
| **基本事件流** | 1. 用户进入个收藏页面 2. 选择查看内容（默认展示收藏图片） 3. 点击图片查看详细内容 4. 退出收藏页面 |
| **扩展事件流** | 2a、选择查看收藏风格：  2a1、点击风格查看详细内容  2a2、回到事件4 |
| **后置条件** | 退出收藏内容查看页面 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **删除收藏** |
| **用例编号** | 02\_05 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户在收藏页面中删除收藏内容 |
| **前置条件** | 用户已正确登录系统 |
| **基本事件流** | 1. 用户进入个收藏页面 2. 选择查看内容（默认展示收藏图片） 3. 长按图片跳出删除按钮 4. 点击按钮删除收藏图片 5. 删除成功 6. 退出收藏页面 |
| **扩展事件流** | 2a1、选择查看收藏风格  2a2、长按风格跳出删除按钮  2a3、点击删除按钮删除收藏风格  2a4、回到事件5 |
| **后置条件** | 退出收藏内容查看页面 |
| **注释** |  |

4）图像处理用例

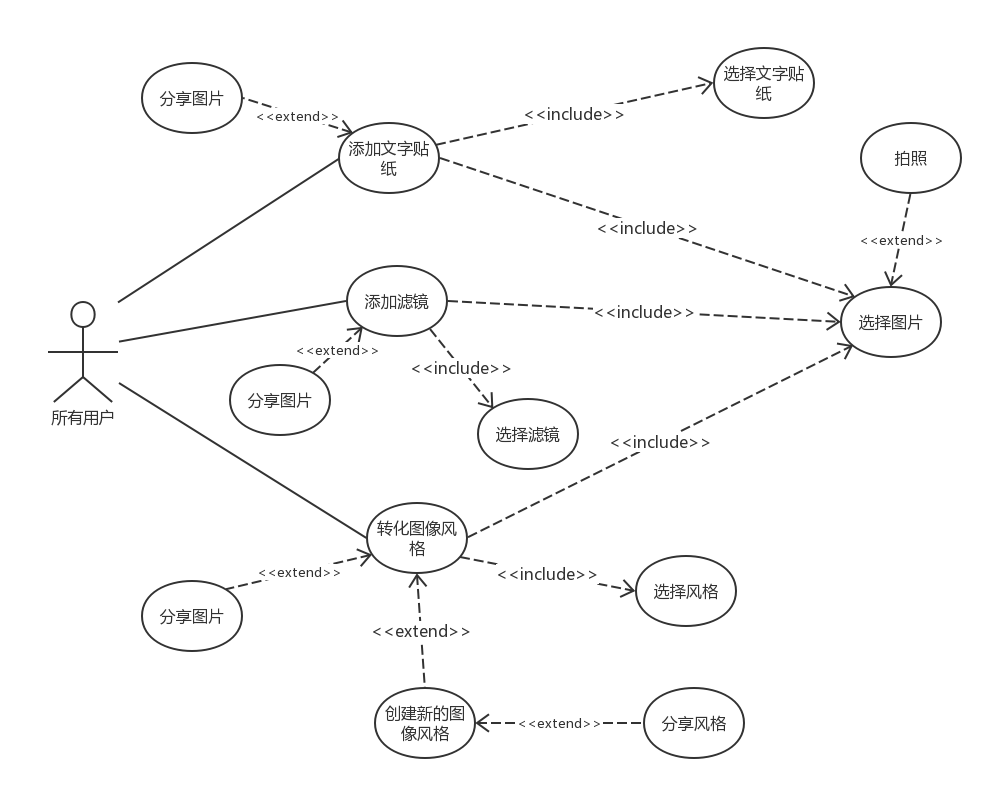


图 2‑0‑4

|  |  |
| --- | --- |
| **用例名称** | **添加文字贴纸** |
| **用例编号** | 03\_01 |
| **参与者** | 所有用户 |
| **简要说明** | 用户选取照片，添加贴纸图片或文字，保存或分享图片 |
| **前置条件** | 用户进入图像处理界面 |
| **基本事件流** | 1. 用户点击贴纸文字按钮 2. 用户选择图片 3. 用户选择贴纸文字 4. 调整贴纸文字大小、颜色等属性 5. 用户保存图片 |
| **扩展事件流** | 2a、用户选择拍照选取图片：  2a1、用户打开摄像头  2a2、用户选择贴纸文字  2a3、调整贴纸文字大小、颜色等属性  2a4、用户点击拍照按钮  2a5、拍照成功，进入事件5  5a、分享图片：  5a1、用户点击分享按钮  5a2、用户选择分享路径  5a3、用户点击提交按钮  5a4、分享成功，并保存图片、结束事件 |
| **后置条件** | 图片保存成功 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **添加滤镜** |
| **用例编号** | 03\_02 |
| **参与者** | 所有用户 |
| **简要说明** | 用户选取照片，添加滤镜，保存或分享图片 |
| **前置条件** | 用户进入图像处理界面 |
| **基本事件流** | 1. 用户点击滤镜按钮 2. 用户选择图片 3. 用户选择滤镜种类 4. 调整滤镜参数属性 5. 用户保存图片 |
| **扩展事件流** | 2a、用户选择拍照选取图片：  2a1、用户打开摄像头  2a2、用户选择滤镜种类  2a3、调整滤镜参数  2a4、用户点击拍照按钮  2a5、拍照成功，进入事件5  5a、分享图片：  5a1、用户点击分享按钮  5a2、用户选择分享路径  5a3、用户点击提交按钮  5a4、分享成功，并保存图片、结束事件 |
| **后置条件** | 图片保存成功 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **转化图像风格** |
| **用例编号** | 03\_03 |
| **参与者** | 所有用户 |
| **简要说明** | 用户选取照片，选取风格并转化图片风格并保存或分享图片 |
| **前置条件** | 用户进入图像处理界面 |
| **基本事件流** | 1. 用户点击风格转化按钮 2. 用户选择图片 3. 用户选择风格种类 4. 用户保存图片 |
| **扩展事件流** | 2a、用户选择拍照选取图片：  2a1、用户打开摄像头  2a2、用户选择风格种类  2a3、用户点击拍照按钮  2a4、拍照成功，进入事件5  3a、用户创建新风格：  3a1、用户点击添加新风格按钮  3a2、用户输入新风格名称  3a3、用户点击添加风格基准图按钮  3a4、用户打开相册并选择风格基准图  3a5、用户点击添加需转化图片按钮  3a6、用户选择图片  3a7、用户预览效果图  3a8:、保存风格，并返回事件1  4a、分享图片：  5a1、用户点击分享按钮  5a2、用户选择分享路径  5a3、用户点击提交按钮  5a4、分享成功，并保存图片、结束事件  3a7a、分享风格：  3a7a1、用户选点击分享风格按钮  3a7a2、分享成功，并返回事件3a8 |
| **后置条件** | 图片保存成功 |
| **注释** |  |

5）图片社交用例

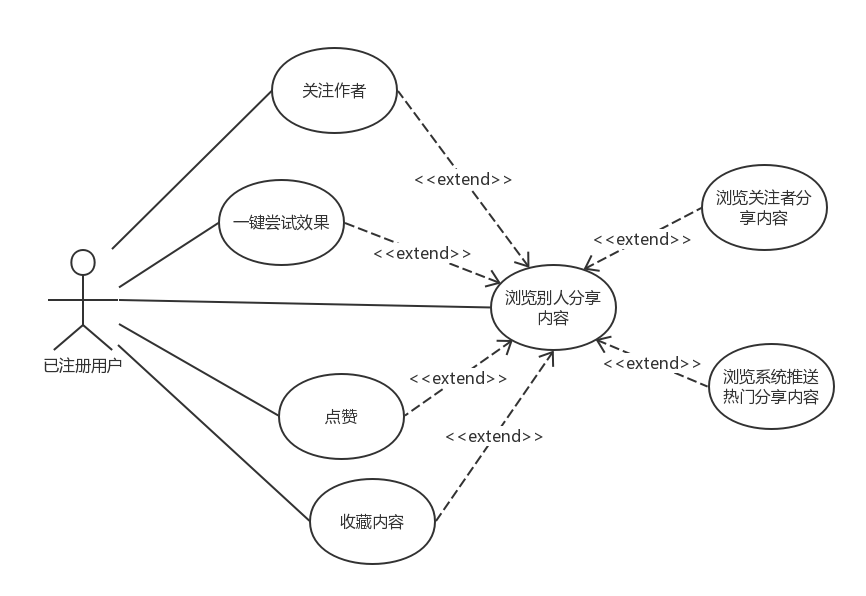


图 2‑0‑5

|  |  |
| --- | --- |
| **用例名称** | **浏览图片、风格** |
| **用例编号** | 04\_01 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户进入社交页面浏览查看其它用户分享内容 |
| **前置条件** | 用户已注册并且已成功登录 |
| **基本事件流** | 1. 用户进入社交页面 2. 用户滑动屏幕浏览内容 3. 用户退出社交界面 |
| **扩展事件流** | 2a、点击朋友圈浏览关注着的分享内容  2b、点击热门内容浏览系统智能推送内容 |
| **后置条件** | 无 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **关注作者** |
| **用例编号** | 04\_02 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户进入社交页面浏览查看其它用户分享内容并关注喜欢的内容的作者 |
| **前置条件** | 用户已注册并且已成功登录 |
| **基本事件流** | 1. 用户进入社交页面 2. 用户滑动屏幕浏览内容 3. 用户点击喜欢内容查看作者 4. 点击关注作者按钮 5. 关注成功 |
| **扩展事件流** | 3a、点击喜欢的图片  3b、点击喜欢的风格 |
| **后置条件** | 关注成功 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **一键尝试效果** |
| **用例编号** | 04\_03 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户浏览其它用户分享的图片或风格，并对喜欢的风格或图片效果进行尝试 |
| **前置条件** | 用户已注册并且已成功登录 |
| **基本事件流** | 1. 用户进入社交页面 2. 用户滑动屏幕浏览内容 3. 用户点击想要感兴趣的风格或图片效果 4. 用户点击试一试按钮 5. 用户选择图片 6. 用户查看效果 |
| **扩展事件流** | 无 |
| **后置条件** | 效果转化成功 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **点赞分享内容** |
| **用例编号** | 04\_04 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户浏览其他用户分享并点赞喜欢的内容 |
| **前置条件** | 用户已注册并以成功登陆 |
| **基本事件流** | 1. 用户进入社交页面 2. 用户滑动屏幕浏览内容 3. 用户点击喜欢的分享内容面板上的点赞按钮 4. 点赞成功 |
| **扩展事件流** | 无 |
| **后置条件** | 点赞成功 |
| **注释** |  |

|  |  |
| --- | --- |
| **用例名称** | **收藏分享内容** |
| **用例编号** | 04\_05 |
| **参与者** | 已注册用户 |
| **简要说明** | 已注册用户浏览其他用户分享并收藏喜欢的内容 |
| **前置条件** | 用户已注册并以成功登陆 |
| **基本事件流** | 1. 用户进入社交页面 2. 用户滑动屏幕浏览内容 3. 用户点击喜欢的分享内容面板上的收藏按钮 4. 收藏成功 |
| **扩展事件流** | 无 |
| **后置条件** | 收藏成功 |
| **注释** |  |

##### (2) 活动图

根据用例图与功能模块图，针对各个细致的功能绘制出活动图，体现出用户和各系统模块之间的活动交互。如下图：

1）登录注册

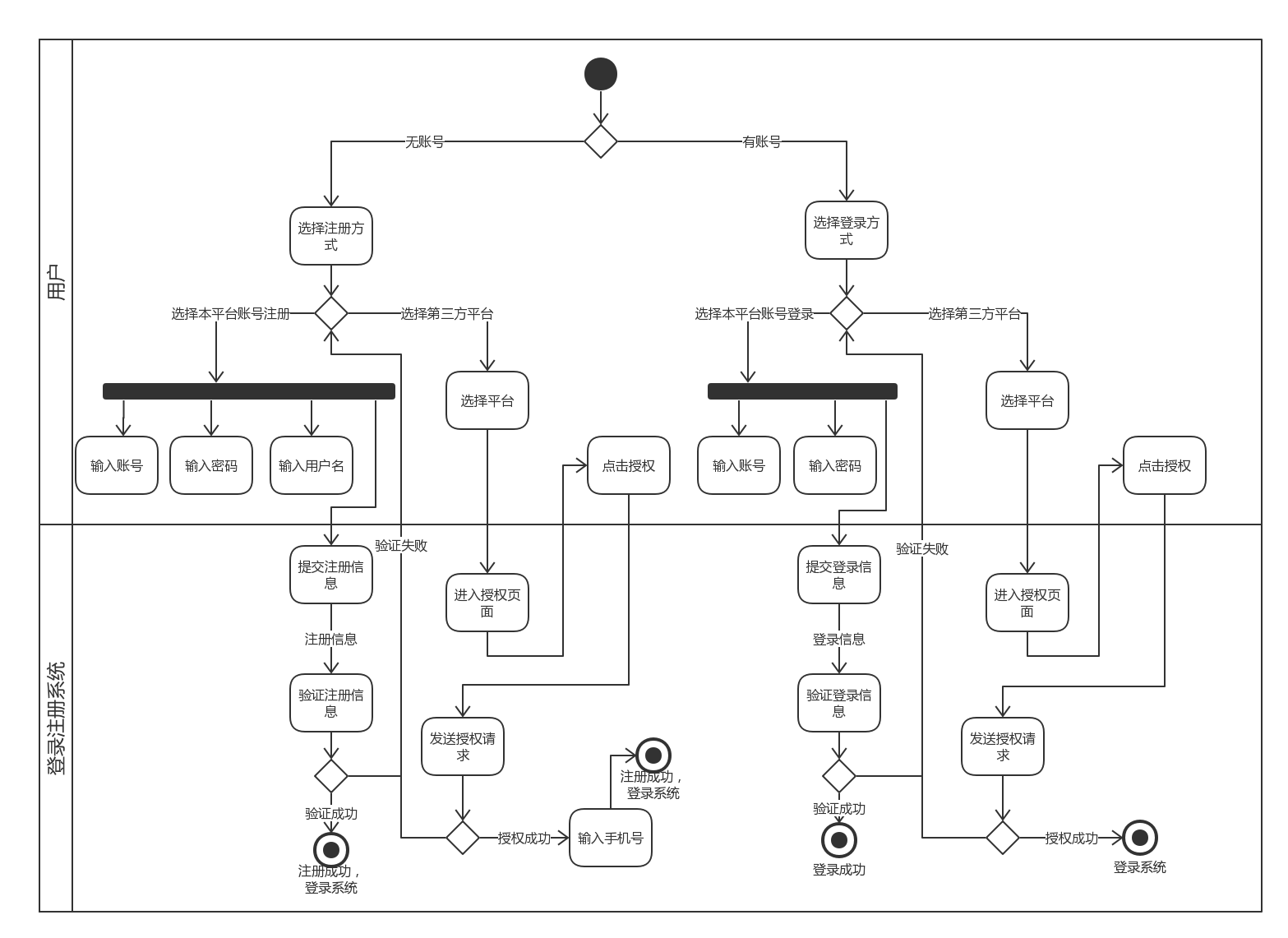


图 2‑0‑6

1. 当用户没有账号，需要进入用户系统时，需要注册账号，注册账号用户可以选择直接在本平台使用手机号进行注册，注册成功后，自动登录系统；用户也可以选择使用第三方平台账号进行注册，系统跳转第三方账号授权页面，用户点击授权之后，注册成功，需要用户输入手机号以完善信息，输入完成后自动登陆系统。
2. 当用户已有账号。需要进入用户系统时。需要登陆账号，根据注册时选择的方式，用户可以选择本平台的账号密码进行登录，验证成功则可登录系统；也可以选择已注册第三方平台账号登录，系统跳转授权页面，用户点击授权，系统登录成功。

2）管理关注列表

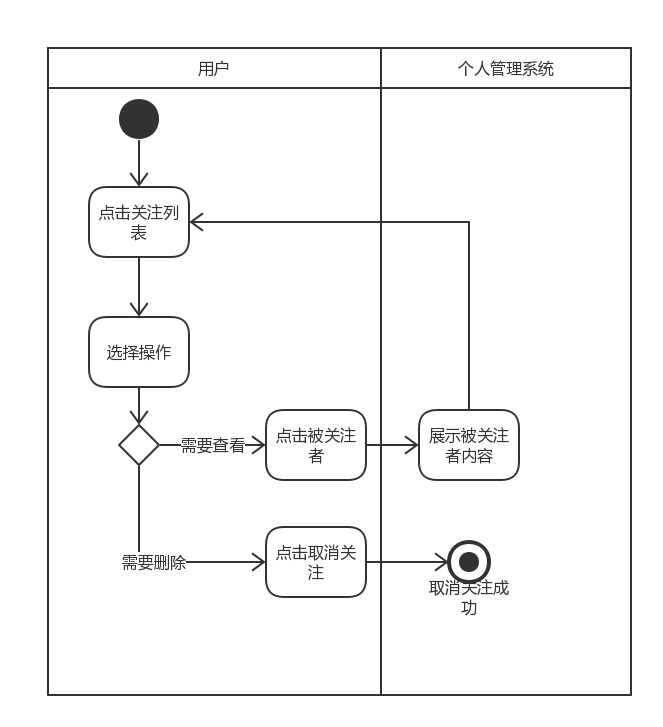


图 2‑0‑7

用户在个人管理系统页面，点击关注列表，可以查看关注的人，用户点击关注的人的头像，系统将展示关注的人的内容；用户右滑关注的人，可出现取消关注选项，用户点击取消关注，可以取消对此人的关注。

3）管理内容

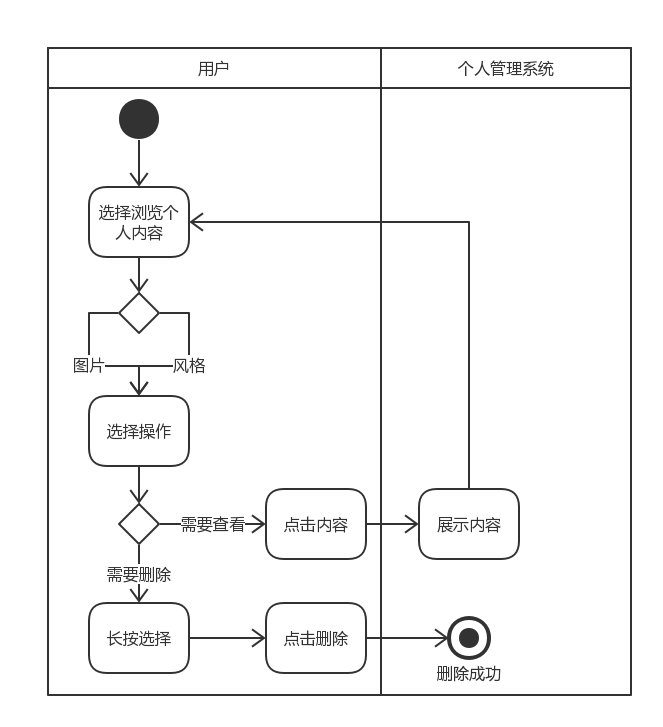


图 2‑0‑8

用户进入个人管理系统页面，默认展示用户在平台上保存过的图片，用户点击风格按钮，切换到展示风格的界面，用户点击内容，系统展示详细内容，用户长按内容，弹出删除菜单，用户点击删除便可以删除内容。

4）管理收藏

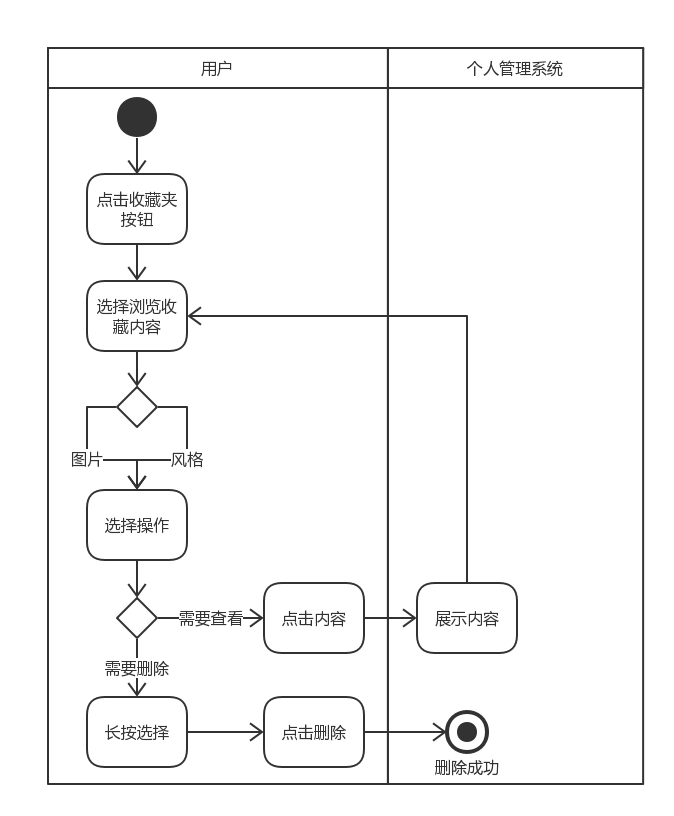


图 2‑0‑9

用户进入个人管理里系统页面，点击收藏夹按钮，进入收藏页面，默认展示用户在平台上收藏过的图片，用户点击风格按钮，切换到展示收藏风格的界面，用户点击内容，系统展示详细内容，用户长按内容，弹出删除菜单，用户点击删除便可以删除收藏内容。

5）修改个人资料

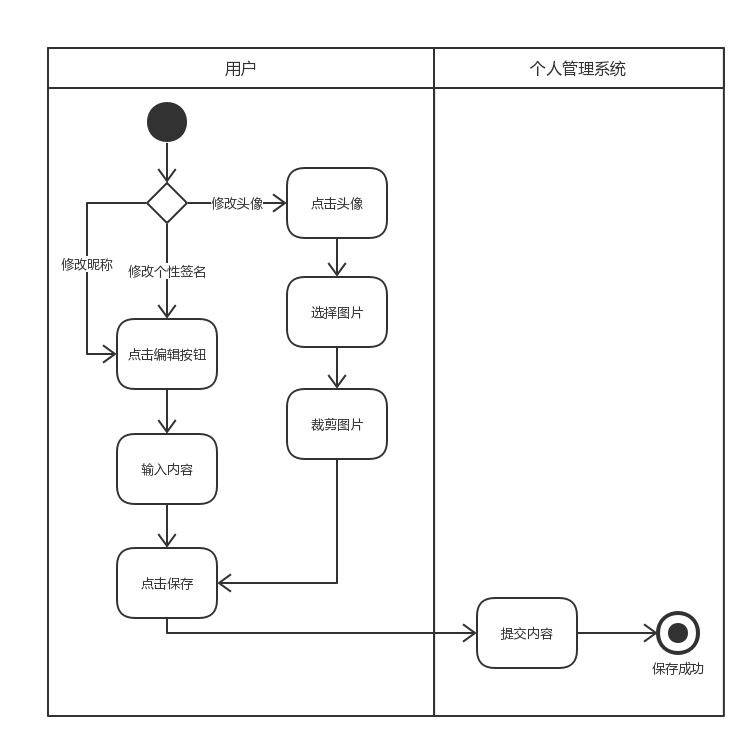


图 2‑0‑10

用户进入个人管理里系统页面，可以修改头像、昵称和个性签名。点击点击头像，系统显示相册，用户选择图片，并裁剪图片为固定尺寸，点击保存，系统提交修改内容并保存成功；用户点击昵称编辑按钮，昵称变为可编辑状态，用户输入内容，点击保存系统提交修改内容并保存成功；用户点击个性签名编辑按钮，个性签名变为可编辑状态，用户输入内容，点击保存系统提交修改内容并保存成功。

6）风格转化

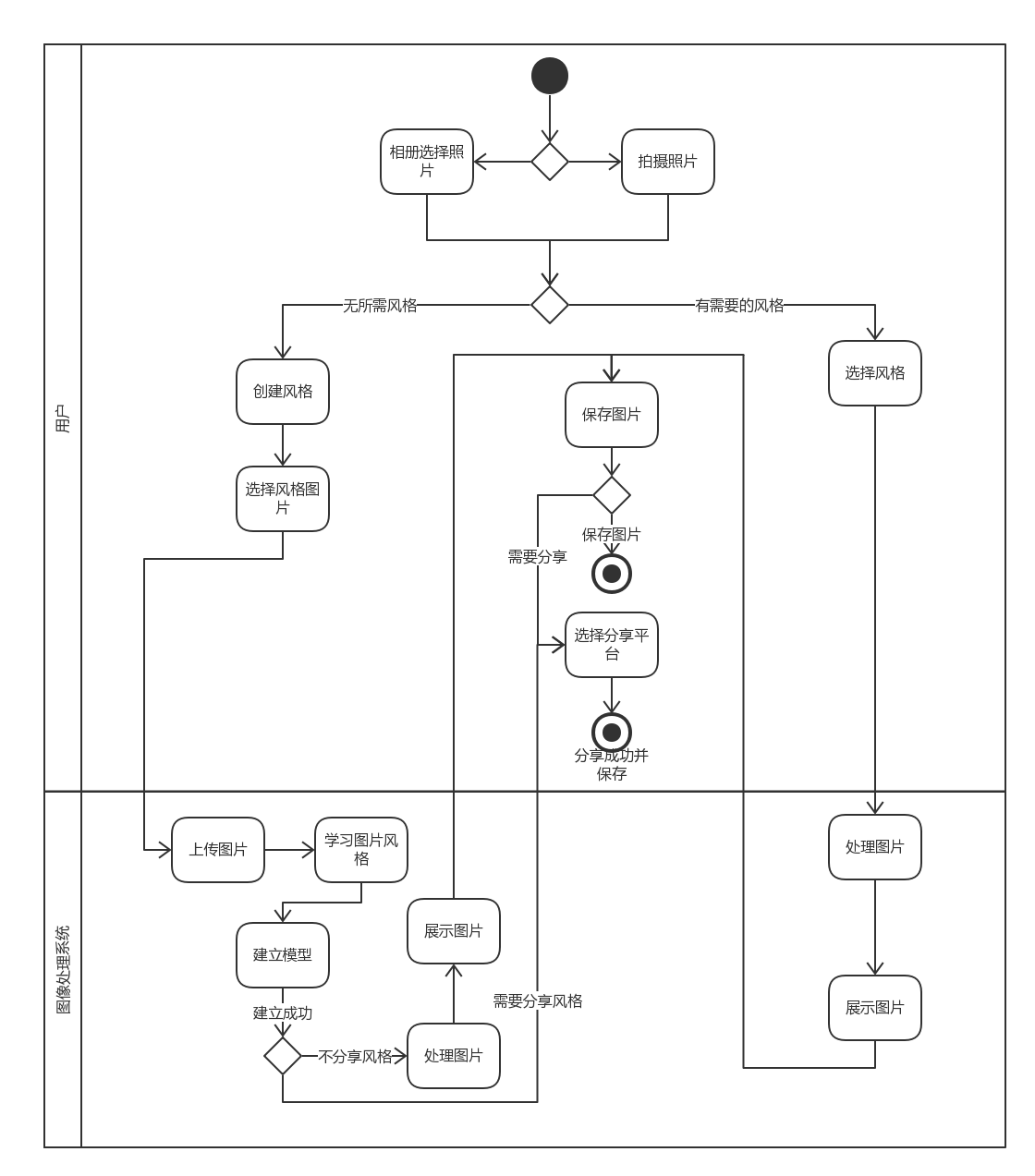


图 2‑0‑11

用户进入风格转化页面，可以选择拍摄照片进行处理也可以选择已有照片进行处理，选择好照片之后，用户选择转化风格。当系统有用户所需风格时，用户选择风格，系统便开始处理图片，处理完成展示图片，用户选择保存图片或分享图片，分享图片可选择分享在该平台，也可选择分享在第三方平台；当系统没有用户所需风格时，用户可以选择创建风格，用户打开相册选择风格图片，系统上传图片，进行风格学习，并建立模型，模型建立成功后，系统处理原图风格转化，并展示图片，用户创建的风格可以进行分享。

7）滤镜处理

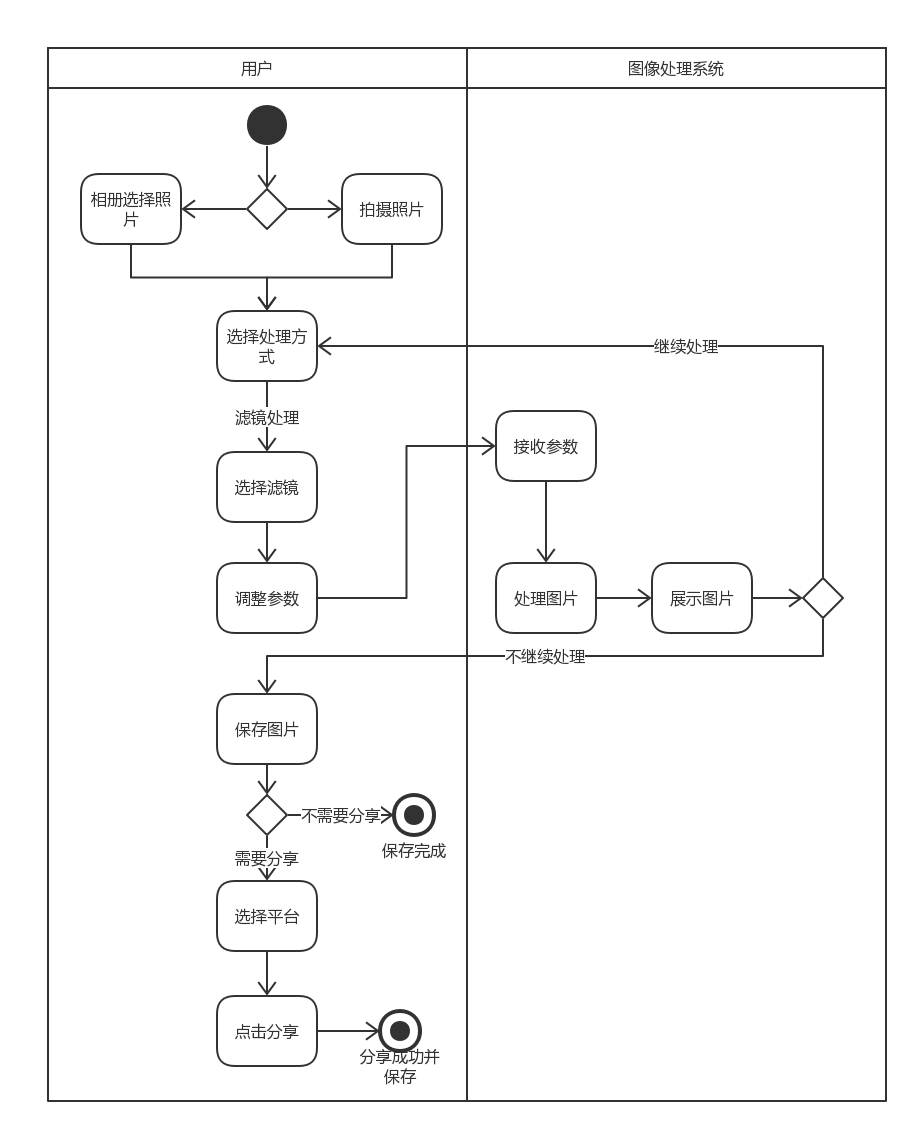


图 2‑0‑12

用户进入图片处理页面，可以选择拍摄照片进行处理也可以选择已有照片进行处理，选择好照片之后，用户选择滤镜并调整滤镜参数，系统接受参数，并实时处理图片，处理完成并展示图片，用户可以选择继续处理图片，也可以选择保存图片，当用户需要分享图片时，需要选择分享平台，点击分享并分享成功。

8）贴纸文字

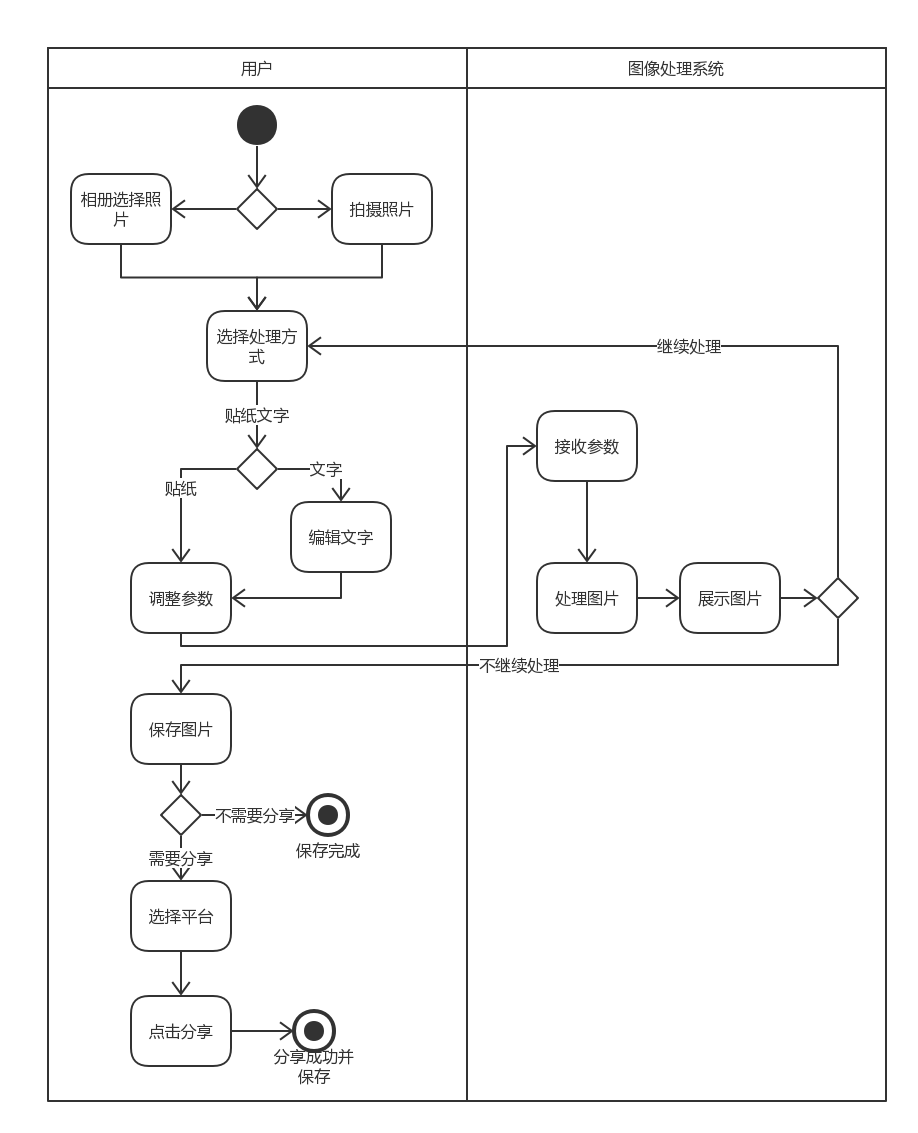


图 2‑0‑13

用户进入图片处理页面，可以选择拍摄照片进行处理也可以选择已有照片进行处理，选择好照片之后，用户选择贴纸图片或者文字，选择文字，用户可以编辑文字，编辑完成后和图片贴纸一样进行参数调整，系统接收参数并处理图片，处理完成后展示图片，用户可以选择继续处理图片，也可以选择保存图片，当用户需要分享图片时，需要选择分享平台，点击分享并分享成功。

9）点赞

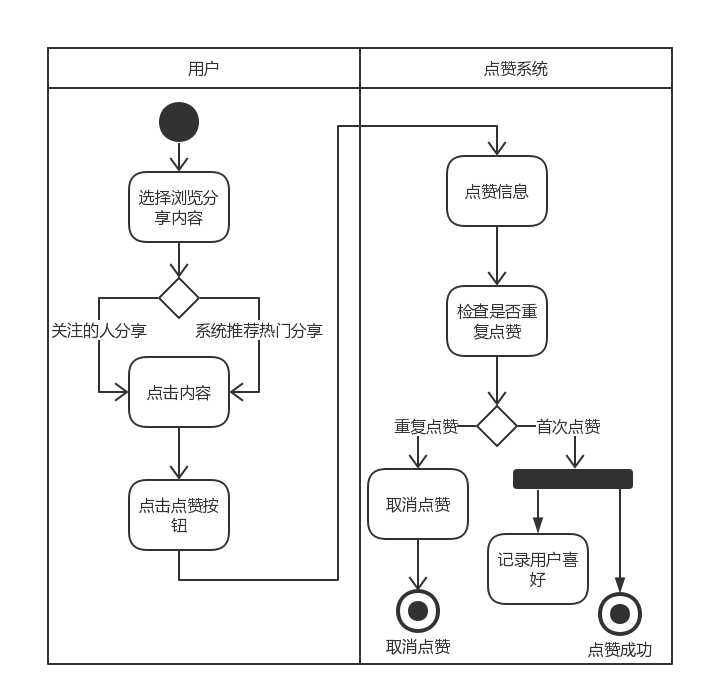


图 2‑0‑14

用户选择浏览分享内容，可选择关注的人的内容，也可选择系统的热门推荐内容，用户点击内容进入内容详情页，用户点击点赞按钮，系统获取点赞信息，检查是否已点赞，如果用户已点赞，则取消之前的点赞，如果用户并未点赞，则点赞成功，系统并记录下用户的喜好。

10）关注

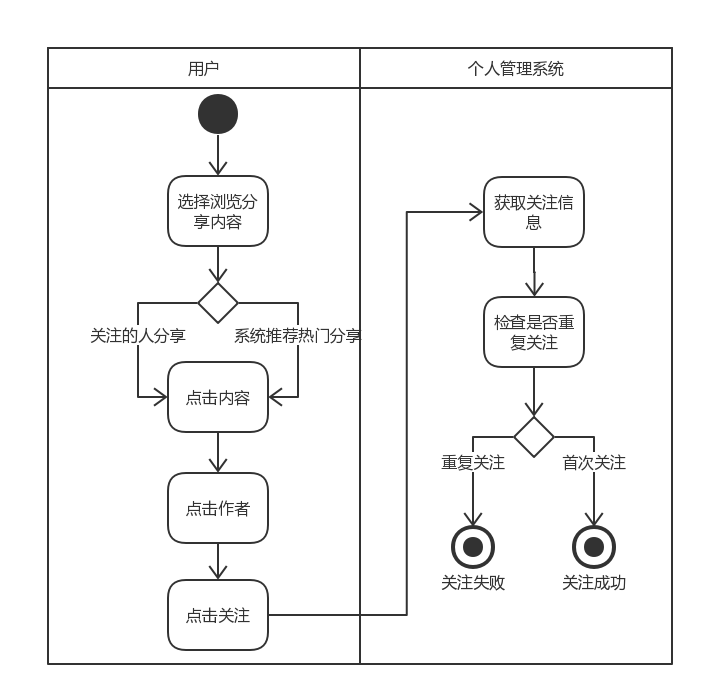


图 2‑0‑15

用户选择浏览分享内容，可选择关注的人的内容，也可选择系统的热门推荐内容，用户点击内容进入内容详情页，详情页有作者头像展示，用户点击作者进入作者主页，用户点击关注，系统获取关注信息，检查是否重复关注，重复关注则关注失败，首次关注则关注成功。

11）收藏内容

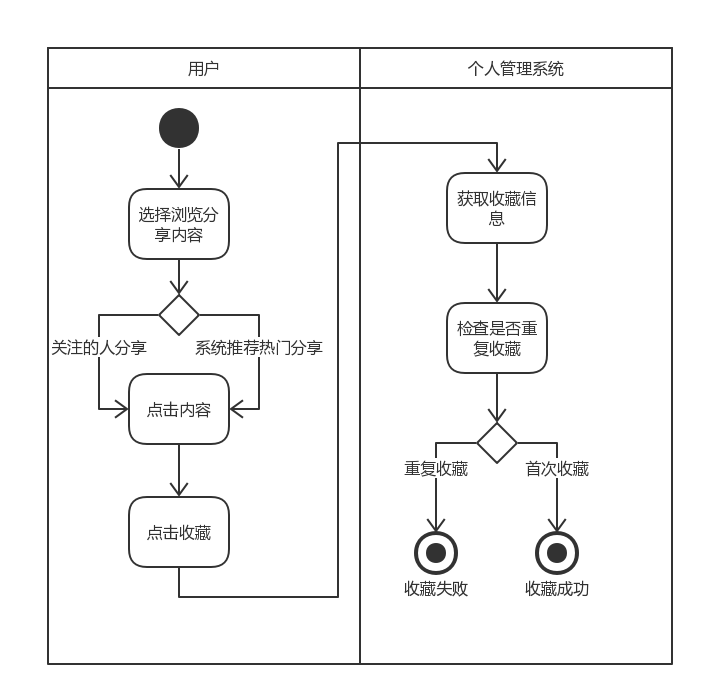


图 2‑0‑16

用户选择浏览分享内容，可选择关注的人的内容，也可选择系统的热门推荐内容，用户点击内容进入内容详情页，用户点击收藏按钮，系统获取收藏信息，系统个检查是否重复收藏，重复收藏，则收藏失败，首次收藏则收藏成功。

12）一键尝试

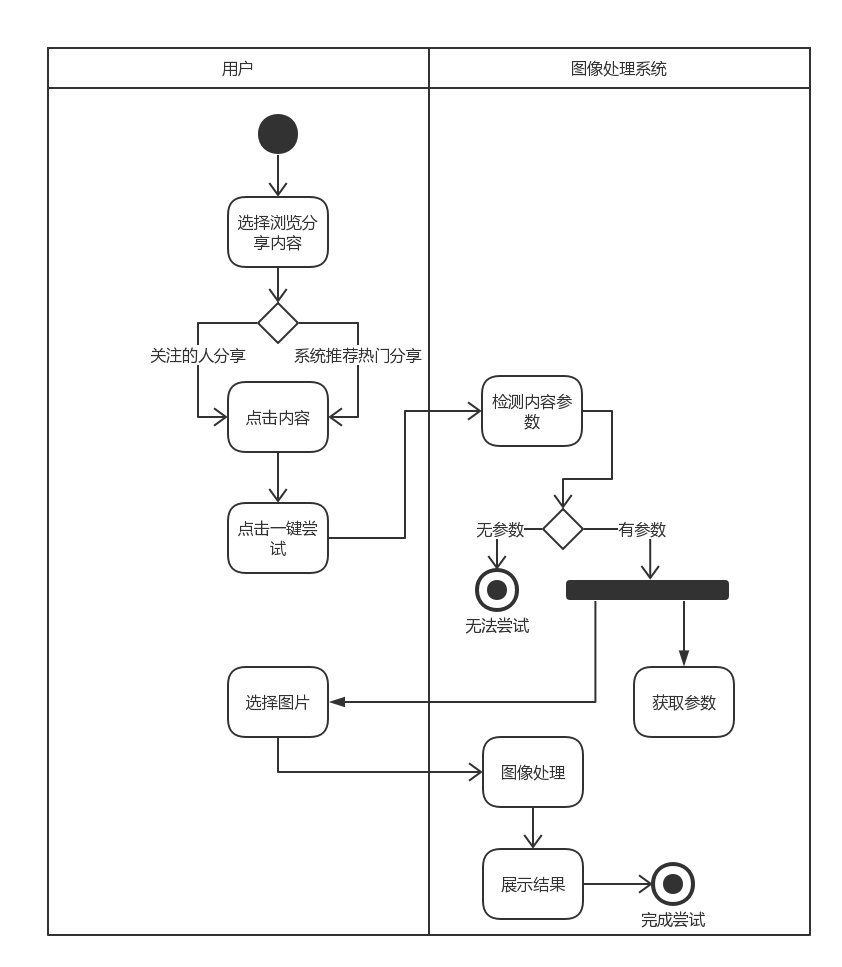


图 2‑0‑17

用户选择浏览分享内容，可选择关注的人的内容，也可选择系统的热门推荐内容，用户点击内容进入内容详情页，用户点击一键尝试按钮，系统检测该内容参数，如果处理无参数，则无法尝试，如果有处理参数，用户选择图片进行尝试吗，选择图片后，系统进行图像处理，处理之后展示图片，并完成尝试。

##### (3) 类图

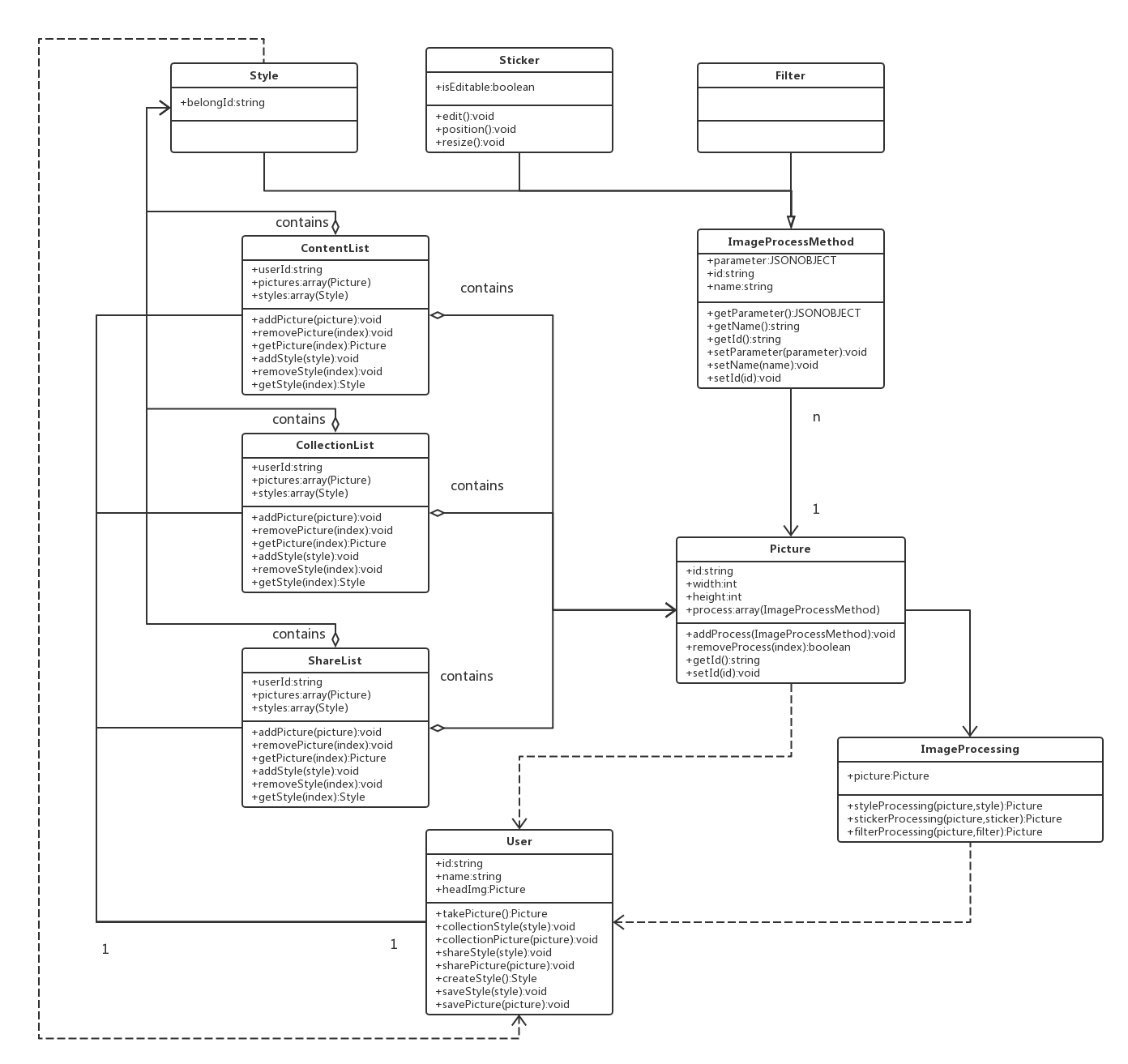


图 2‑18

### 2.1.2非功能性需求分析

根据用户对本系统的要求，确定系统在响应时间、可靠性、安全等方面有较高的性能要求。

**（一）界面需求**

系统的界面要求如下：

* 1. 页面内容：主题突出，站点定义、术语和行文格式统一、规范、明确，栏目、菜单设置和布局合理，传递的信息准确、及时。内容丰富，文字准确，语句通顺；专用术语规范，行文格式统一规范。
  2. 导航结构：页面具有明确的导航指示，且便于理解，方便用户使用。
  3. 艺术风格：界面、版面形象清新悦目、布局合理,字号大小适宜、字体选择合理，前后一致，美观大方；动与静搭配恰当,动静效果好；色彩和谐自然,与主题内容相协调。

**（二）响应时间需求**

* + 1. 系统响应时间：为满足用户对系统高效要求，以及系统运行的高流畅性要求，系统尽可能的优化APP代码，使APP在网络通讯正常的情况下，系统响应时间在1~2秒之间。
    2. 网络响应时间：在网络通讯速度得到保障的情况下，用户从发送指令到服务器到设备执行命令的响应时间为1秒。例如在20M宽带WIFI或4G网络的条件下，响应速度可达到1~2秒。
    3. 图像处理响应时间：图像处理是该系统运行时间最长的一项处理，为了让用户得到更好的体验，当用户选择已有风格进行处理，响应时间将控制在一分钟以内，当用户选择没有的风格进行处理时，响应时间控制在一小时以内。

1. 其他需求
   * + - 1. 开放性需求：系统应具有十分的灵活性，以适应将来功能扩展的需求。
2. 可扩展性需求：系统设计要求能够体现扩展性要求，以适应将来功能扩展的需求。

## 2.2项目管理

### 2.2.1项目生命周期与组织

（1）项目生命周期

本项目开发经历了团队人员组织、可行性分析、需求分析、概要设计、详细设计、系统实现、测试、反馈维护阶段。团队人员组织，选择项目经理为团队创建的第一步，一名合格的项目经理应当具备广博的知识、丰富的经历、良好的协调能力、良好的职业道德、良好的沟通与表达能力、良好的领导能力等要求，接下来的整个项目开发阶段均依据项目经理的指示循序渐进，整个过程中，项目经理扮演的是“一把手”的角色，自始至终都发挥“一把手”的作用，定期进行汇报和交流，以获取支持、理解和资源的调配。可行性分析阶段主要从技术可行性、操作可行性、经济可行性等几方面对项目的可行性进行分析，并提出可行性方案，此项目开发是一个长期的、有风险的、耗时长的工程项目，因此在进行正式的系统开发之前，要从有益性、可能性和必要性三个方面对未来系统的经济效益、社会效益做初步分析，避免盲目投资、减少不必要的损失，最后编写可行性报告。需求分析阶段通过阅读需求文档，公司网站了解信息，向有过车间工作管理经验的人士请教，查阅纸质及电子资料等多种途径了解需求信息。概要设计阶段根据需求分析获得的信息按照功能进行模块划分，建立模块的层级结构及调用关系、确定模块间的接口及人机界面等。详细设计阶段对概要设计进行细化，详细设计每个模块实现算法，所需的局部结构。系统实现主要指软件系统的编码与实现。

（2）项目组织

a. 项目组成员对不同的角色做了相应的分析，从而做到了项目制作的规范化管理，也从自身项目的管理进一步了解项目组要做出的项目的业务逻辑流程。

b. 结合小组成员擅长技能、兴趣爱好等因素，以混合方案组织团队，进行角色和职责细化安排。

c. 制定人员配置管理计划。

### 2.2.2角色与职责

|  |  |  |
| --- | --- | --- |
| **姓名** | **角色** | **职责** |
| 阿勋 | 项目经理 | 1. 项目整体规划 2. 项目策划设计 3. 项目进度安排 4. 项目管理 5. 系统架构设计 |
| 阿鑫 | 技术经理 | 1. 算法设计 2. 技术开发 3. 系统结构设计 |
| 阿云 | 技术经理 | 1. 技术开发 2. 服务器研究 3. API设计 4. 数据库设计 |
| 阿帅 | 前端经理 | 1. APP开发 2. 前端技术研究 3. 模块设计 |
| 小红 | UI设计师 | 1. 界面设计 2. 优化交互 |

### 2.2.3项目评审

在项目研发过程中为了保证研发进度及研发产品的质量，会安排大量的评审活动。

本系统在开发过程中通过项目评审作为项目管理的一部分来对研发过程和产品质量的保证。评审记录如表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 工作产品 | 评审方式 | 评审时间 | 评审人员 |
| 需求说明书 | 正式 | 2017-6-20 | 全体成员及指导老师 |
| 概要设计说明书 | 正式 | 2017-7-1 | 全体成员及指导老师 |
| 文档质量评审 | 正式 | 2017-7-20 | 全体成员及指导老师 |
| 测试用例 | 正式 | 2017-7-25 | 全体成员及指导老师 |
| 项目总结 | 正式 | 2017-7-29 | 全体成员及指导老师 |

### 2.1.4任务分解与安排进度

分阶段描述各阶段主要工作任务、时间范围、参与人员及工作成果等。项目任务分解与进度安排表如下表所示:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 任务名称 | 开始时间 | 结束时间 | 参与人员 | 工作量（人/天） | 工作成果 |
| 需求分析阶段 | 2017-6-1 | 2017-6-20 | 阿勋 | 20 | 项目管理计划、需求规格说明书。 |
| 技术准备阶段 | 2017-6-10 | 2017-7-1 | 阿鑫、阿云、阿帅 | 60 | 编码规范、技术说明。 |
| 详细设计阶段 | 2017-6-20 | 2017-6-25 | 阿勋、小红 | 10 | 项目开发计划、  质量保证计划，UI设计。 |
| 系统设计阶段 | 2017-6-26 | 2017-6-30 | 阿勋、阿鑫、阿云、小红 | 20 | 概要设计、数据库设计、详细设计、测试计划 |
| 系统编码阶段 | 2017-7-1 | 2017-7-20 | 阿鑫、阿云、阿帅 | 60 | 源代码、可执行与安装文件。 |
| 系统测试阶段 | 2017-7-21 | 2017-7-26 | 阿勋、阿鑫、阿帅 | 15 | 功能测试、集成测试、系统测试、非功能测试、测试文档、用户操作手册、安装说明。 |
| 系统修正阶段 | 2017-7-26 | 2017-8-1 | 阿勋、阿鑫、阿帅、阿云 | 20 | 源代码定稿、安装可执行与文件定稿、安装说明定稿、用户操作手册定稿。 |

### 2.2.4任务分解与进度安排

### 2.2.5项目过程管理

### 2.2.6项目里程碑管理

### 2.2.7项目质量管理

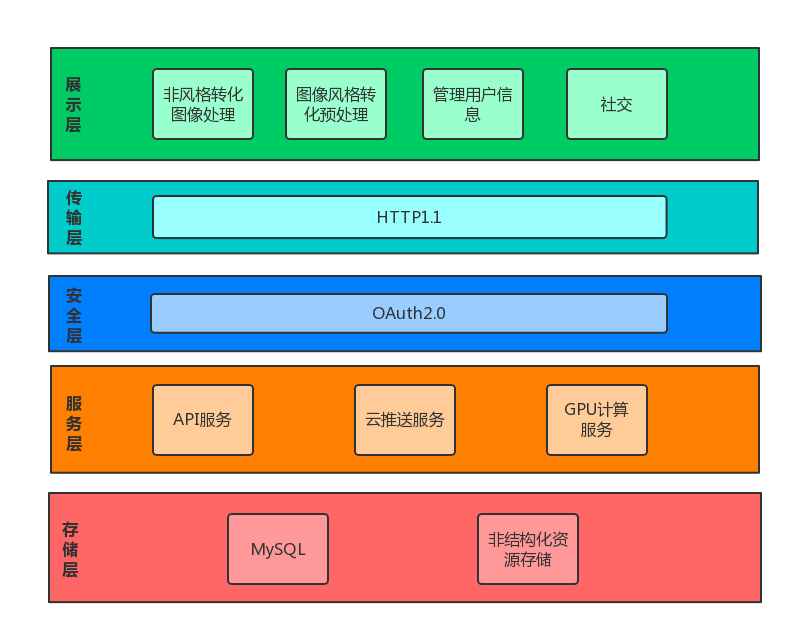
### 2.2.8项目沟通管理

### 2.2.9项目风险管理

# 三、技术路线及实现方案

## 3.1项目概要设计

### 3.1.2系统架构设计



系统架构如图所示，为C/S架构，分为展示层（APP）、传输层、安全层、服务层、存储层，各层详细内容如下所示：

1. 展示层：
2. 非风格转化处理：滤镜处理和图片贴纸处理均在展示层（APP）上进行。
3. 风格转化预处理：图像风格转化之前的用户预处理工作，选区、裁剪、人像轮廓识别均在展示层（APP）上进行。
4. 管理用户信息：包括登录注册、查看分享内容。
5. 社交：包括点赞、关注、收藏。
6. 传输层：采用HTTP1.1协议进行通讯，主要使用协议中的POST、GET、PUT、DELETE方法。
7. 安全层：使用OAuth2.0协议作为用户登录的认证。用户登录网站获取一个令牌，每粗发送请求时将令牌放至Header中，以供服务器进行验证身份。
8. 服务层：
9. API服务：系统架构为C/S架构，所有请求服务均建立在HTTP之上，系统制定并编写一套符合Restful规则的API接口，以供客户端获取资源使用，API服务使用Python的Flask框架进行开发。
10. 云推送服务：使用第三方推送平台，当用户风格转化完成之后，系统调用第三方推送系统将消息推送至客户端，以提醒用户完成转化。
11. GPU计算：图像风格转化需要进行大量计算，光靠CPU进行计算远远不够，因此需要GPU进行计算，GPU在图像矩阵方面的计算速率大约是同等级别CPU的60倍左右，因此图像风格转化选用GPU计算。由于每次计算时间至少在1分钟以上，并且每次风格转化均为独占设备，所以不能实时响应用户的转化需求，故需要队列等候。GPU计算所使用的计算框架为基于Tensorflow计算引擎的Keras框架。
12. 存储层：
13. MySQL：MySQL用于存储关系型数据，使用SQL语句对数据库进行增、 删、改、查等操作。
14. 非结构化资源存储：系统将会保存大量图像资源，因此需要专门的存储空间对图像资源进行存储，转化为静态资源，访问图像资源则是以URL的形式进行访问。

### 3.1.3系统接口设计

图酷系统的系统架构为B/S架构，故前端APP与后端服务器之间的交互采用Restful API的规范进行交互，通信协议为HTTP1.1。如下为关键API接口的设计：

1. API设计
   1. 获取用户基本信息

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/baseinfo | |
| Method | GET | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户id |
| **传递参数** | | |
| 无 | | |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| headImageUrl | String | 头像图片 |
| userName | String | 用户名 |
| signature | String | 用户个性签名 |
| 说明 | 返回实例请见注解a | |

* 1. 获取用户分享图片信息

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/sharecontents/pictures | |
| Method | GET | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| lastRequestTime | String | 上次请求时间 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| imgUrl | String | 图片URL |
| imgId | String | 图片id |
| processValue | JSONObjectString | 图片处理的参数，详情请看注解b，如果没有就为空 |
| styleId | String | 风格id，如果没有就为空 |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| 说明 | 返回实例请见注解c | |

* 1. 获取用户分享风格信息

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/sharecontents/styles | |
| Method | GET | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| lastRequestTime | String | 上次请求时间 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| styleName | String | 风格名称 |
| styleId | String | 风格id |
| styleImgUrl | String | 风格图片URL |
| resultImgUrl | String | 结果图片URL |
| originImgUrl | String | 原图片URL |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| 说明 | 返回实例请见注解d | |

* 1. 获取用户所有图片信息

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/allcontents/pictures | |
| Method | GET | |
| URL参数 | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| 传递参数 | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| lastRequestTime | String | 上次请求时间 |
| 返回参数 | | |
| 参数名称 | 参数类型 | 参数说明 |
| imgUrl | String | 图片URL |
| imgId | String | 图片id |
| processValue | JSONObjectString | 图片处理的参数，详情请看注解b，如果没有就为空 |
| styleId | String | 风格id，如果没有就为空 |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| isShared | Boolean | 是否分享过 |
| 说明 | 返回实例请见注解f | |

* 1. 获取用户所有风格信息

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/allcontents/styles | |
| Method | GET | |
| URL参数 | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| 传递参数 | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| lastRequestTime | String | 上次请求时间 |
| 返回参数 | | |
| 参数名称 | 参数类型 | 参数说明 |
| styleName | String | 风格名称 |
| styleId | String | 风格id |
| styleImgUrl | String | 风格图片URL |
| resultImgUrl | String | 结果图片URL |
| originImgUrl | String | 原图片URL |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| isShared | Boolean | 是否分享过 |
| 说明 | 返回实例请见注解g | |

* 1. 获取朋友圈社交内容图片列表

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userid}}/momentcontents/pictures | |
| Method | GET | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| lastRequestTime | String | 上次请求时间 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| imgUrl | String | 图片URL |
| imgId | String | 图片id |
| processValue | JSONObjectString | 图片处理的参数，详情请看注解b），如果没有就为空 |
| styleId | String | 风格id，如果没有就为空 |
| authorInfo | String | 作者的用户基本信息 |
| isFollowed | Boolean | 是否已关注 |
| isLiked | Boolean | 是否已点赞 |
| isCollected | Boolean | 是否已收藏 |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| 说明 | 返回实例请见注解h | |

* 1. 获取朋友圈社交内容风格列表

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userid}}/momentcontents/styles | |
| Method | GET | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| lastRequestTime | String | 上次请求时间 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| styleName | String | 风格名称 |
| styleId | String | 风格id |
| styleImgUrl | String | 风格图片URL |
| resultImgUrl | String | 结果图片URL |
| originImgUrl | String | 原图片URL |
| authorInfo | String | 作者的用户基本信息 |
| isFollowed | Boolean | 是否已关注 |
| isLiked | Boolean | 是否已点赞 |
| isCollected | Boolean | 是否已收藏 |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| 说明 | 返回实例请见注解g | |

* 1. 获取收藏图片列表

|  |  |  |
| --- | --- | --- |
| Path | /user/{{userId}}/collections/pictures | |
| Method | GET | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| imgUrl | String | 图片URL |
| imgId | String | 图片id |
| processValue | JSONObjectString | 图片处理的参数，详情请看注解b），如果没有就为空 |
| styleId | String | 风格id，如果没有就为空 |
| authorInfo | String | 作者的用户基本信息 |
| isFollowed | Boolean | 是否已关注 |
| isLiked | Boolean | 是否已点赞 |
| isCollected | Boolean | 是否已收藏 |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| 说明 | 返回实例请见注解h | |

* 1. 获取收藏风格列表

|  |  |  |
| --- | --- | --- |
| Path | /user/{{userId}}/collections/styles | |
| Method | GET | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户分享图片过多时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| styleName | String | 风格名称 |
| styleId | String | 风格id |
| styleImgUrl | String | 风格图片URL |
| resultImgUrl | String | 结果图片URL |
| originImgUrl | String | 原图片URL |
| authorInfo | String | 作者的用户基本信息 |
| isFollowed | Boolean | 是否已关注 |
| isLiked | Boolean | 是否已点赞 |
| isCollected | Boolean | 是否已收藏 |
| likeNum | Int | 点赞数量 |
| collectNum | Int | 收藏数量 |
| 说明 | 返回实例请见注解i | |

* 1. 获取关注列表

|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/follows | |
| Method | GET | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| pageIndex | Int | 用户关注过多用户时，需要pageIndex参数分页请求，默认每次请求30条内容 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| follows | JSONObjectString | 关注者列表，包含关注者基本信息 |
| 说明 | 返回实例请见注解j | |

* 1. 上传图片

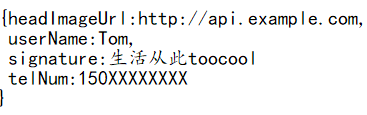
|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/allcontents/pictures | |
| Method | POST | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| imgUrl | String | 图片URL |
| processValue | JSONObjectString | 图片处理的参数，详情请看注解b），如果没有就为空 |
| styleId | String | 风格id，如果没有就为空 |
| isShared | Boolean | 是否分享 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| isSuccess | Boolean | 上传成功 |
| 说明 | 上传实例请见注解l | |

* 1. 上传风格

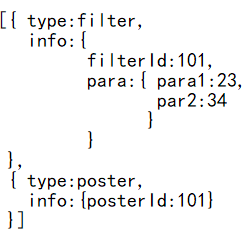
|  |  |  |
| --- | --- | --- |
| Path | /users/{{userId}}/allcontents/styles | |
| Method | POST | |
| **URL参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| userId | String | 用户Id |
| **传递参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| styleName | String | 风格名称 |
| styleImgUrl | String | 风格图片URL |
| resultImgUrl | String | 结果图片URL |
| originImgUrl | String | 原图片URL |
| isShared | Boolean | 是否分享 |
| **返回参数** | | |
| 参数名称 | 参数类型 | 参数说明 |
| isSuccess | Boolean | 上传成功 |
| 说明 | 上传实例请见注解m | |

(2) 注解

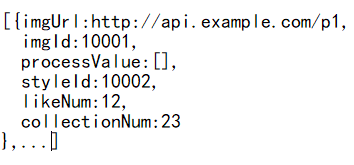
* 1. 用户基本信息实例如下：



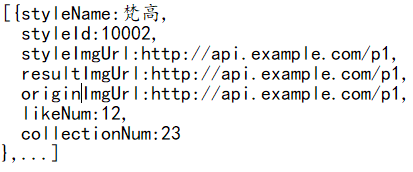
* 1. 图片处理参数完整参数实例如下：



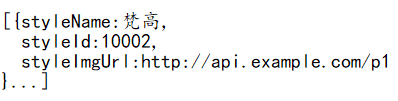
* 1. 用户分享图片信息返回实例如下：



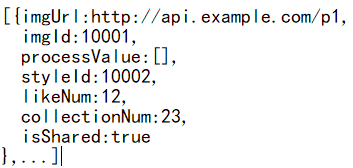
* 1. 用户分享风格信息返回实例如下：



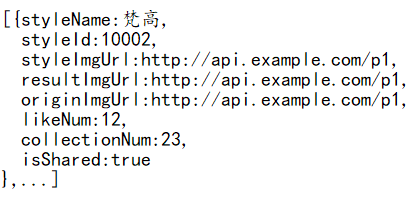
* 1. 获取系统风格返回实例如下：



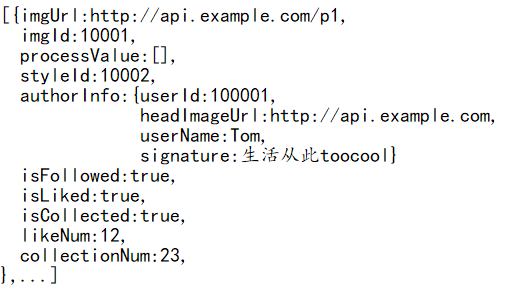
* 1. 获取用户所有图片信息返回实例如下：



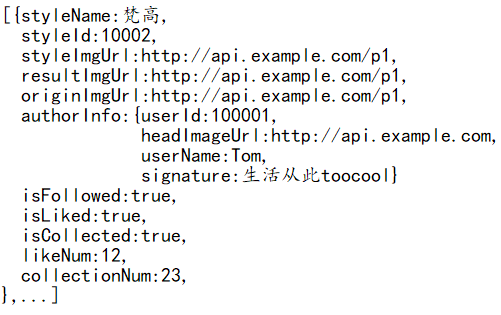
* 1. 获取用户所有风格信息返回实例如下：



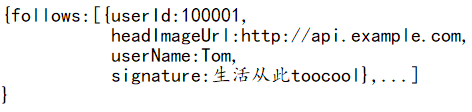
* 1. 获取朋友圈社交内容图片列表、获取系统推荐社交内容图片列表、获取收藏图片列表的返回实例如下：



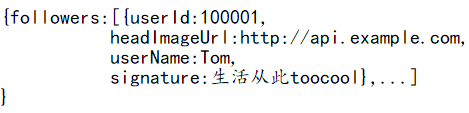
* 1. 获取朋友圈社交内容风格列表、获取系统推荐社交内容风格列表、获取收藏风格列表的返回实例如下：



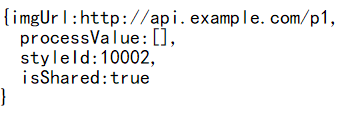
* 1. 获取关注列表返回实例如下：



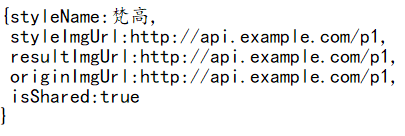
* 1. 获取粉丝列表返回实例如下：



* 1. 上传图片上传参数实例如下：

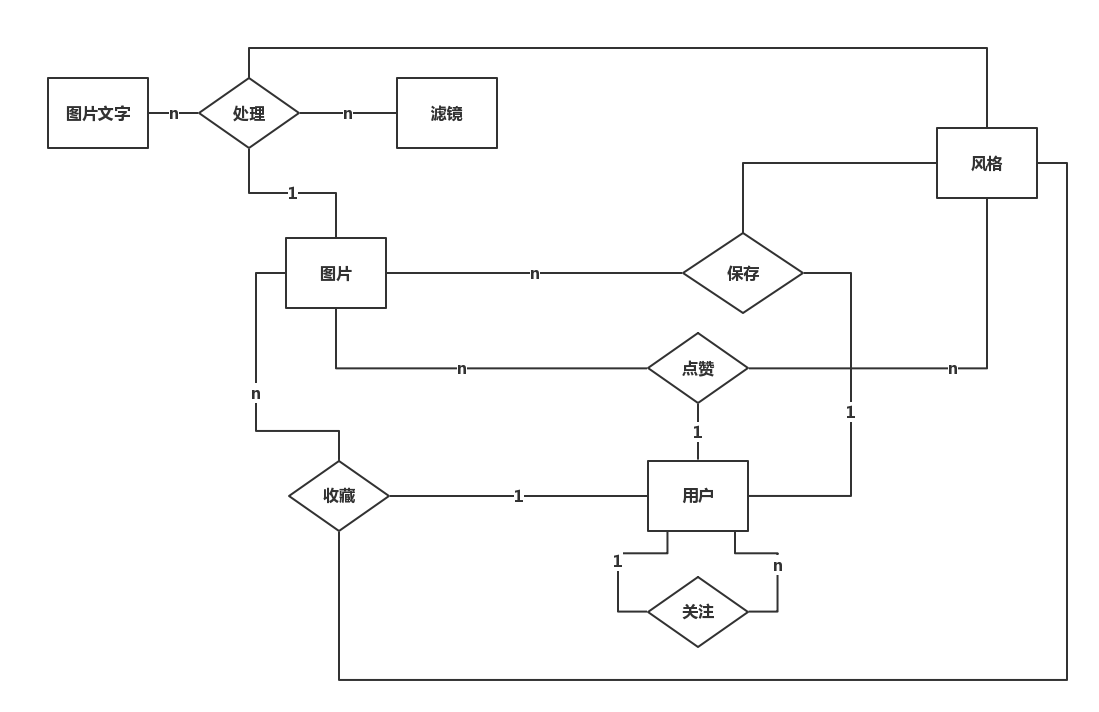


* 1. 上传风格上传参数实例如下：



## 3.2数据库设计

（1） E-R图



图酷系统实体关系图如上所示，实体有图片文字、滤镜、风格、图片、用户，关系有处理关系、收藏关系、点赞关系、关注关系、保存关系，具体关系描述如下所示：

1. 系统需要维护用户信息、图片信息、风格信息、贴纸文字信息、滤镜信息。
2. 用户上传处理的照片以及用户创建的风格，用户可以选择保存，一名用户可以创建多个风格、处理多个图片并保存，保存时可选择是否分享。
3. 用户浏览社区精选内容时，可选择关注其创作者，一个用户可以关注多个用户。
4. 用户浏览别人分享的内容时，可以收藏喜欢的内容，一个用户可以收藏多个图片或者风格。
5. 一张图片可以由多个滤镜、多个图片贴纸处理，但是一张图片只能由一种风格进行处理。
6. 用户浏览别人分享内容时，可以对喜欢的内容进行点赞，一个用户可以点赞多个图片或风格。

（2） 根据上述关系描述，设计如下数据项（主键用下划线，外键用双下划线表示）：

1. 用户：用户id，手机号，用户名，个性签名，头像，时间戳；
2. 关注列表：用户id，关注用户id，时间戳；
3. 收藏列表：用户id，内容id，内容作者用户id，时间戳；
4. 保存列表：用户id，内容id，是否分享，时间戳；
5. 点赞列表：用户id，内容id，点赞用户id，时间戳；
6. 图片：图片id，图片位置，滤镜参数，贴纸文字参数，风格id，时间戳；
7. 风格：风格id，风格图位置，图位置，原图位置，风格名称，是否为系统默认，时间戳；
8. 滤镜：滤镜id，滤镜名称；
9. 贴纸文字：贴纸文字id，贴纸文字名称；

（3） 根据上述数据项，设计如下数据词典：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UserList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| userId | String | 从10000递增 | 无 | 用户id | 否 |
| telNum | String | 11个字符 | 无 | 手机号 | 否 |
| userName | String | 无 | TK+手机号后5位 | 用户名 | 否 |
| signature | String | 30字以内 | 生活从此toocool | 个性签名 | 否 |
| headerImgUrl | String | 无 | 无 | 用户头像链接 | 可 |
| createTime | Timestamp | 无 | 无 | 用户创建账号时间 | 否 |
| 主键 | userId | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FollowList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| userId | String | 无 | 无 | 用户id | 否 |
| followId | String | 无 | 无 | 关注的用户id | 否 |
| followTime | Timestamp | 无 | 无 | 用户关注时间 | 否 |
| 主键 | userId、followId | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CollectionList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| userId | String | 无 | 无 | 用户id | 否 |
| imgId | String | 无 | 无 | 图片id | 可 |
| styleId | String | 无 | 无 | 风格id | 可 |
| authorId | String | 无 | 无 | 作者id | 否 |
| collectTime | Timestamp | 无 | 无 | 收藏时间 | 否 |
| 主键 | userId | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ContentList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| userId | String | 无 | 无 | 用户id | 否 |
| imgId | String | 无 | 无 | 图片id | 可 |
| styleId | String | 无 | 无 | 风格id | 可 |
| isShare | Boolean | 无 | 无 | 是否分享 | 否 |
| saveTime | Timestamp | 无 | 无 | 收藏时间 | 否 |
| 主键 | userId | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **LikeList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| userId | String | 无 | 无 | 用户id | 否 |
| contentId | String | 无 | 无 | 内容id，可为图片id或风格id | 可 |
| listTime | Timestamp | 无 | 无 | 点赞时间 | 否 |
| 主键 | userId、contentId | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PictureList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| pictureId | String | 以字符1结尾 | 10001 | 图片id，从10000开始递增 | 否 |
| pictureUrl | String | 无 | 无 | 图片URL | 否 |
| filterValue | JSONString | 无 | 无 | 滤镜参数 | 可 |
| posterValue | JSONString | 无 | 无 | 贴纸文字参数 | 可 |
| styleId | String | 无 | 无 | 风格id | 可 |
| createTime | Timestamp | 无 | 无 | 创建时间 | 否 |
| 主键 | pictureId | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **StyleList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| styleId | String | 以字符2结尾 | 10002 | 风格id，从10000开始递增 | 否 |
| styleName | String | 无 | 无 | 风格名称 | 否 |
| styleImgUrl | String | 无 | 无 | 风格图片URL | 否 |
| originImgUrl | String | 无 | 无 | 原图URL | 否 |
| resultImgUrl | String | 无 | 无 | 结果图片URL | 否 |
| isSystemDefult | Boolean | 无 | False | 是否为系统默认 | 否 |
| createTime | Timestamp | 无 | 无 | 创建时间 | 否 |
| 主键 | styleId | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FilterList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| filterId | String | 无 | 无 | 滤镜id | 否 |
| filterName | String | 无 | 无 | 滤镜名称 | 否 |
| 主键 | filterId | | | | |

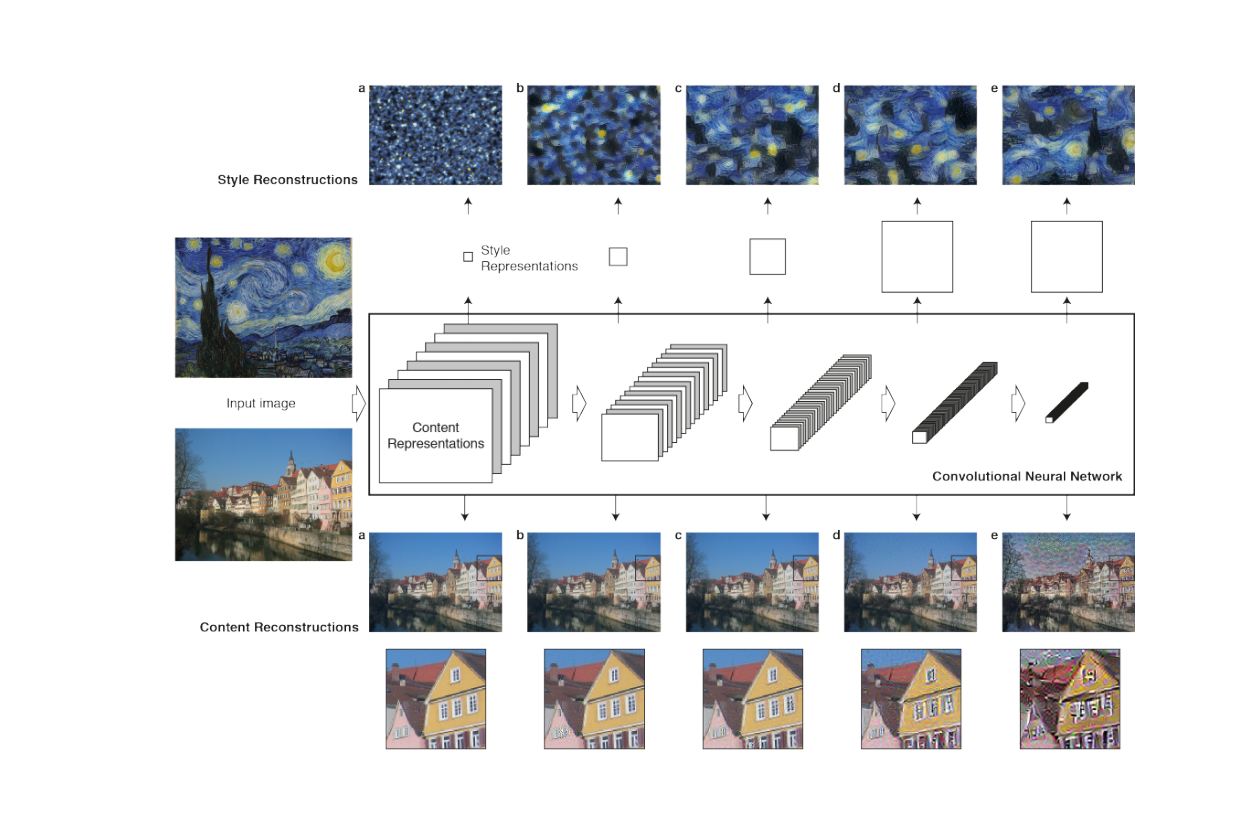
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PosterList** | | | | | |
| 数据项 | 类型 | 约束 | 默认值 | 说明 | 可否为空 |
| posterId | String | 无 | 无 | 贴纸文字 | 否 |
| posterName | String | 无 | 无 | 贴纸文字名字 | 否 |
| 主键 | posterId | | | | |

## 3.3技术难点实现

### 3.3.1 风格迁移的实现

（1）风格迁移的本质与原理：

风格迁移的本质就是将一张图片的材质、质地应用于另外一张图片上。然而图片的质地与材质则可以理解为不同的颜色、线条、纹理组合在一起给人视觉感受。图像风格迁移原理则是运用卷积神经网络将风格图片的质地进行捕捉，同时也对内容图片中的内容进行捕捉，然后将两者应用于一张噪声图上，通过不断的修改调整噪声图的内容，使其具有内容图片的内容，又具有风格图片的质地材质，最终将得到一张风格迁移的结果图,原理图如下所示：



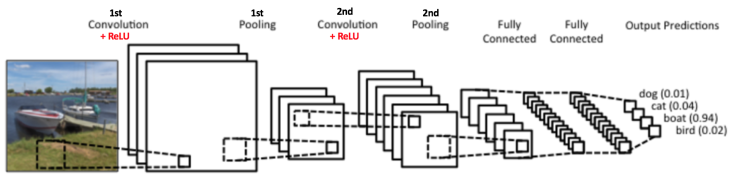
（2）卷积神经网络：

卷积神经网络，简称为CNN，CNN的基本结构包括两层，其一为特征提取层，每个神经元的输入与前一层的局部接受域相连，并提取该局部的特征。一旦该局部特征被提取后，它与其它特征间的位置关系也随之确定下来；其二是特征映射层，网络的每个计算层由多个特征映射组成，每个特征映射是一个平面，平面上所有神经元的权值相等。特征映射结构采用sigmoid函数作为卷积网络的激活函数，使得特征映射具有位移不变性。此外，由于一个映射面上的神经元共享权值，因而减少了网络自由参数的个数。卷积神经网络中的每一个卷积层都紧跟着一个用来求局部平均与二次提取的计算层，这种特有的两次特征提取结构减小了特征分辨率。

此次风格迁移，我们并未搭建新的卷积神经网络，而是使用google的卷积神经网络VGG，如下图所示为VGG系列的结构图：



其中VGG16和VGG19在图像风格迁移方面都有较好的表现，所以本次项目我们选择使用VGG16作为卷积神经网络，总共有16层神经网络，由卷积层（conv）、激活函数（relu）、池连接（pool）交替组成，其中卷积层我们选用全连接卷积、激活函数选用sigmoid函数、池连接选用最大化池连接（Max Pooling），原理图如下所示：



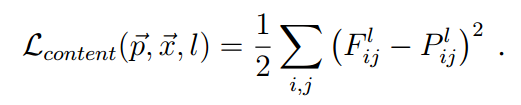
本系统虽然选用的神经网络为VGG16，但是并没有使用到VGG16的所有层次，在风格图片的特征集合的捕捉上，我们使用了VGG16中的五层网络，分别为conv1\_1、conv2\_1、conv3\_1、conv4\_1、conv5\_1；而内容图片的内容捕捉方面，则使用了VGG16的一层网络，为conv5\_2；其选择的依据是根据我们参考了arxiv.org网站上的一篇名为A Neural Algorithm of Artistic Style的论文而定的，下图为上述几层神经网络所取特征集合的展示：



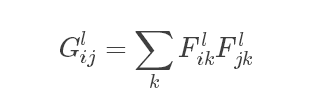
横轴为风格图所占权重，纵轴为风格图所在卷积层，从图中可以明显看出，随着卷积层的增加，风格图片的特点捕捉越来越明显，结果图也越来越具有风格的特点。

（3）计算过程

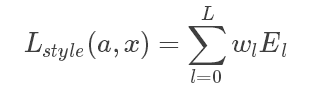
 在CNN 中, 假设某一 layer 含有  个 filters, 那么将会生成  个 feature maps，每个 feature map 的维度为 ，  是 feature map 的高与宽的乘积。所以每一层 feature maps 的集合可以表示为 , 表示第 i个 filter在 position j 上的 activation。所以，我们可以给出 content 的 cost function:



为了建立风格的representation，我们先利用 Gram matrix 去表示每一层各个 feature maps 之间的关系，  ,   是 feature maps i , j 的内积：



结合所有层，可以得到总的cost:



最后将 content 和 style 的 cost 相结合，最终可以得到:



## 3.4环境搭建

## 3.5界面设计

# 四、成本模型及可行性分析

## 4.1成本模型

软件开发的成本分为直接成本和间接成本两种，相对应的具体内容和费用如表4-1所示：

表4-1图像风格转化系统费用列表

| 成本类别 | 成本内容 | 费用（元） |
| --- | --- | --- |
| 直接成本  （人/年） | 项目经理工资 | 5000\*13=65000 |
| 项目经理社保/公积金 | 年薪\*45%=29250 |
| 程序员 | 3500\*13=45500 |
| 程序员社保/公积金 | 年薪\*45%=20475 |
| 测试员 | 3000\*13=39000 |
| 测试员社保/公积金 | 月薪\*45%=17550 |
| 技术顾问/咨询顾问 | 8000\*13=104000 |
| 技术顾问/咨询顾问  社保/公积金 | 104000\*45%=46800 |
| 电脑(6000元/3年) | 2000 |
| 服务器(30000元/3年) | 10000 |
| 房租(5元/㎡/天) | 5\*365\*150/7=39107 |
| 水电费 | 1460 |
| 电话费 | 1095 |
| 交通费 | 1460 |
| 间接成本  （年） | 管理费(项目报价\*10%) |  |
| 办公费 | 6000 |

根据表4-1所列出的费用表，以40天作为本项目的项目周期，项目经理1位、程序员3位、UI设计1位，可具体计算出整个项目的成本，如表4-2所示：

表4-2图像风格转化系统成本核算

| 职务 | 成本类别 | 成本内容 | 费用(元) |
| --- | --- | --- | --- |
| 项目经理 | 直接成本 | 项目经理工资 | 11397 |
| 项目经理 社保/公积金 | 5129 |
| 电脑 | 225 |
| 服务器 | 1096 |
| 房租 | 4286 |
| 水电费 | 160 |
| 电话费 | 120 |
| 交通费 | 160 |
| 合计 | | | 16375 |
| 程序员 | 直接成本 | 程序员工资 | 7123 |
| 程序员社保/公积金 | 3205 |
| 电脑 | 225 |
| 服务器 | 1096 |
| 房租 | 4286 |
| 水电费 | 160 |
| 电话费 | 120 |
| 交通费 | 160 |
| 合计 | | | 13277 |
| UI设计 | 直接成本 | UI设计工资 | 4986 |
| UI设计 社保/公积金 | 2244 |
| 电脑 | 225 |
| 服务器 | 1096 |
| 房租 | 4286 |
| 水电费 | 160 |
| 电话费 | 120 |
| 交通费 | 160 |
| 合计 | | | 12244 |
| 间接成本 | 办公费 |  | 657 |
| 成本总计 | | | 42453 |

由上表分析可得成本为42453元，项目报价将定在70000元。（该计算结果为商业软件开发成本分析，本系统为竞赛类系统，相关成本将会有一定的折扣）。

## 4.2可行性分析

### 4.2.1市场分析

全球经济正在迅速增长，发展高科技产业，重点在IT业，IT业中首要的是发展软件业。软件产业附加价值高，技术含量大，对于提升国民经济整体科技含量和促进产业结构调整具有突出的意义。尤其是软件产业属于知识密集型产业，对于自然资源的依附度较小，十分符合我国目前的现实条件。

随着智能手机的普及，以及手机拍摄科技的不断提高，人们对美的追求越来越高，人人都爱拍照，拍照之后对照片的美化需求也越来越大，市面上出现了种类繁多的图片美化软件，这些美图软件各有各的特色，以美图APP为例，市面上很多的图片美化软件都以美白、磨皮、瘦脸、亮眼等美颜功能为主，在面部识别处理方面有一定技术含量，其主要针对的用户群体为女性爱拍照爱美的用户。除了美图以外，还有其他的图片美化APP，有以社区为主的、风格小清新的APP，例如黄油相机，这类APP主要以有趣的高质量的推文来吸引用户，并提供了一个较好的平台给用户以展示交流优质的图片，在图像处理方面主推有趣好看的贴纸文字，没有很难大的技术难度。除了以社交、美颜为主的，图片美化软件，还有以技术为主的图片处理软件，例如Snapseed，这是一款号称移动端的PS软件，其很多图像处理都可以和PS相媲美，比较有技术含量，但是缺乏一个平台，让用户可以交流图像处理经验。当然也有较好的比较全面的图片处理软件，例如in，这是一个有社区，有一定技术含量的图片处理软件。

综上图片处理软件市场竞争激烈，需要在此立足，要么是有更好玩有趣吸引人的社区，要么是有更深的处理技术，让一张图片有更多的玩法。而我们的图酷APP则是一款集合了浅度社交、滤镜、贴纸文字、图像风格转化为一体的图像处理软件，社交则是以点赞、收藏、关注、图库为主，浅度的社交没有评论与聊天，没有过多的打扰，让人更能沉浸在自我的世界里；而技术方面则是使用了当下比较火热的卷积神经网络机器学习技术，能够将两张或多张图片进行融合，从而生成一张更具特色的图片。同时，我们的设计理念以酷为主题，针对的用户群体主要以文艺青年、嘻哈青年、朋克青年等年轻人为主。图酷APP与市场上的图片处理软件整体上有一定区别，也有一定的辨识度，所以市场方面具有可行性。

### 4.2.2政策分析

国务院研究部进一步鼓励软件产业和集成电路产业发展的政策措施，包括投融资、税收和市场规范等，将对国内软件产业的发展带来巨大的影响。

（1）《国务院关于印发鼓励软件产业和集成电路产业发展若干政策的通知》

a. 多方筹措资金，加大对软件产业的投入。建立软件产业风险投资机制，鼓励对软件产业的风险投资。由国家扶持，成立风险投资公司，设立风险投资基金。初期国家可安排部分种子资金，同时通过社会走向募股和吸收国内外风险投资基金等方式筹措资金。

b. 为软件企业在国内外上市融资创造条件。尽快开辟证券市场创业板。软件企业不分所有制性质，凡符合证券市场创业上市条件的，应优先给予安排；对具有良好市场前景及人才优势的软件企业，在资产评估中无形资产占净资产的比例可由投资方自行商定；支持软件企业到境外上市融资。经审核符合境外上市资格的软件企业，均可允许到境外申请上市筹资。

c. 支持开发重大共性软件和基础软件。国家科技经费重点支持具有基础性、战略性、前瞻性和重大关键共性软件技术的研究与开发，主要包括操作系统、大型数据库管理系统、网络平台、开发平台、信息安全、嵌入式系统、大型应用软件系统等基础软件和共性软件。属于国家支持的上述软件研究开发项目，应以企业为主，产学研结合，通过公开招标方式，择优选定项目承担者。

（2）《国务院办公厅关于进一步促进服务外包产业发展的复函》

延续并完善示范城市发展服务外包的政策措施，文件中提出：完善服务外包产业发展环境。加强服务外包信息安全法律法规体系建设，推动示范城市所在省（区、市）尽快研究出台服务外包知识产权和信息安全保护等方面的地方法规。开展服务外包信息安全认证评估，引导和支持企业建立内控机制。继续加强服务外包产业基础研究，出台相关行业标准。鼓励地方政府在员工住宿、物业租赁等方面对服务外包企业给予支持。

（3）《 浙江省人民政府关于印发浙江省服务外包产业十二五发展规划的通知》

a. 政策引导、统筹发展。以科学发展观统领服务外包产业发展，充分发挥服务外包对经济转型升级的“绿色引擎”作用，促进全省经济发展方式的转变。

b. 示范带动、引领发展。要充分把握国际服务产业转移重点，结合浙江城市化和服务业发展特点，明确服务外包产业发展的重点区域、重点产业。要充分发挥国家级服务外包示范城市的引领带动作用，选择一些发展潜力大的城市进行重点扶持，促进示范城市与其他城市的协同发展。

c. 产业互动、协调发展。服务外包产业发展要紧密结合全省经济社会发展进程，围绕“十二五”经济社会发展总体目标，将服务外包产业与软件信息业、金融、现代物流、文化创意、咨询、信息、财会、教育、医疗等现代服务业发展密切结合，实现服务外包与先进制造业、现代服务业、新兴战略性产业等互动发展。

d. 人才先行、跨越发展。服务外包产业发展的关键是人才。要制定完善相关扶持政策，支持人才培训和引进。

（4）《浙江省软件和信息服务业十二五发展规划》

《浙江省软件和信息服务业十二五发展规划》指出“十二五”期间，根据产业现实基础和未来发展趋势，到“十二五”末，全省软件和信息服务业务规模继续保持在全国前列，实现业务收入3000亿元，年复合率达到15%以上；实现增加值1200亿元，年复合增长率达到14%以上，占全省生产总值的比重超过3%。构建布局合理，结构优化、竞争有序、创新能力和服务水平一流的产业体系，建成技术先进、功能完善、安全可靠、覆盖城乡的网络基础设施，形成一批知名企业和著名品牌，成为具有国内外重要影响力的软件和信息服务业基地，加快建设软件和信息服务业强省。

因此本项目在政策方面可行。

### 4.2.3经济可行性分析

### 4.2.4技术可行性分析

### 4.2.5操作可行性分析

### 4.2.6法律可行性分析

## 4.3风险管理

### 4.3.1技术风险

### 4.3.2组织风险

### 4.3.3人员流动风险

### 4.3.4控制进度风险

# 附录：会议记录摘要