



西南科技大学毕业答辩

# 公共社交网站的设计与实现

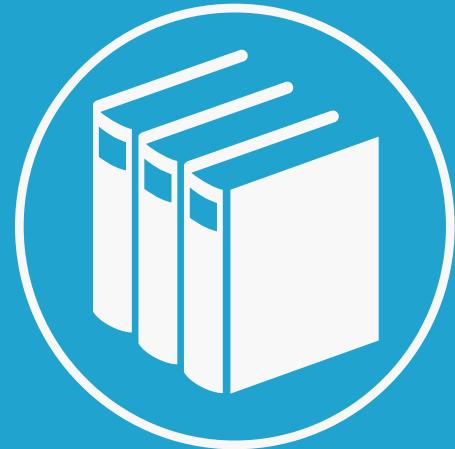
指导老师：王昆  
答辩人：彭磊

院系专业：计算机科学与技术学院  
信息安全





- 1 选题的目的和意义 ➔**
- 2 研究的现状 ➔**
- 3 相关技术 ➔**
- 4 分析与设计 ➔**
- 5 关键点及其实现 ➔**
- 6 成功展示 ➔**



## 一、选题的目的和意义

- 选题背景
  - 目的和意义
- 

互联网给人们的沟通提供了前所未有的便利,使人们足不出户就能进行生产和消费等日常活动,足不出户就能办公、购物、交往等等,它以前所未有的开放性和互动性,每天都在创造着新的工作方式、生活方式和思维方式。而社交网站(Social Network Site)的出现,更好的论释了这个理念。



## 背景

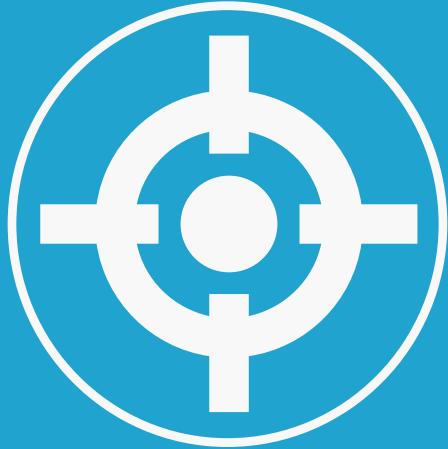
SNS全称**Social Networking Services**, 在中文里叫社会性网络服务。它是互联网应用中以支持用户互动行为为特征的一类**Web2.0**应用和系统的统称。它的发展大体经历了四个阶段：早期概念化阶段——小世界理论，娱乐化阶段，社交图阶段和垂直社交网络应用阶段。在社交网络中，用户之间的交流从传统的以“计算机”为单位变成了以“人”为单位，网络的参与主体发生了根本性的变化。

如今,互联网给人们的沟通提供了前所未有的便利,使人们足不出户就能进行生产和消费等日常活动,足不出户就能办公、购物、交往等等,它以前所未有的开放性和互动性,每天都在创造着新的工作方式、生活方式和思维方式。而社交网站(**Social Network Site**)的出现,更好的论释了这个理念。



## 目的和意义

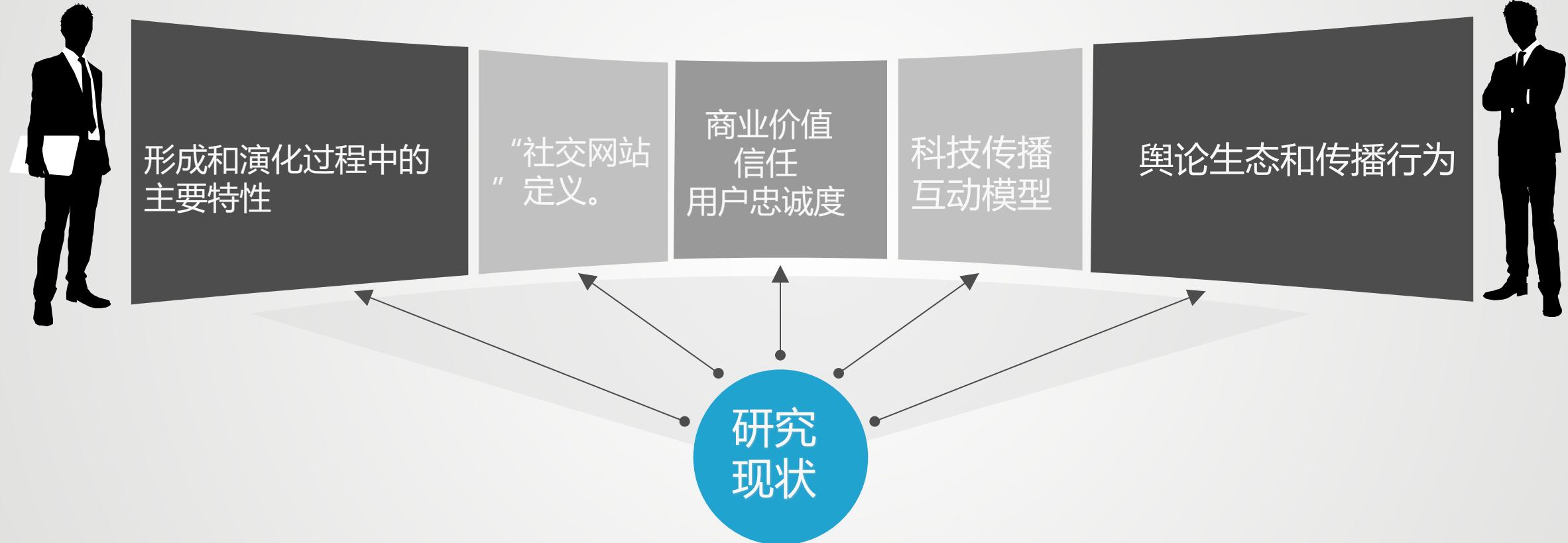
随着社交网络和移动智能设备的迅速发展，人们将越来越多的日常生活信息分享在社交网络上，例如照片、社交关系、行为活动等。此次课题《社交网站的设计与实现》旨在设计与实现一个具有个人色彩的实名制交友网站，以朋友为中心，通过“朋友的朋友”来进行网络社交拓展，安全地真实地友好地扩大自己的交友圈，获取更多的资讯，与好友进行更友好更频繁更多样化的互动，通过网络，将人与人之间的联系变得更加紧密。由于该项目在国内尚且算是一个新兴的服务，我国相关领域对于该类服务研究尚未健全，但是在国外该项目已经存在较多研究成果，所以在该项目中，可以参考他们的研究成果来更加准确的探索国内有关社交网站方面更深层次的领域。



## 二、选题的现状与目标

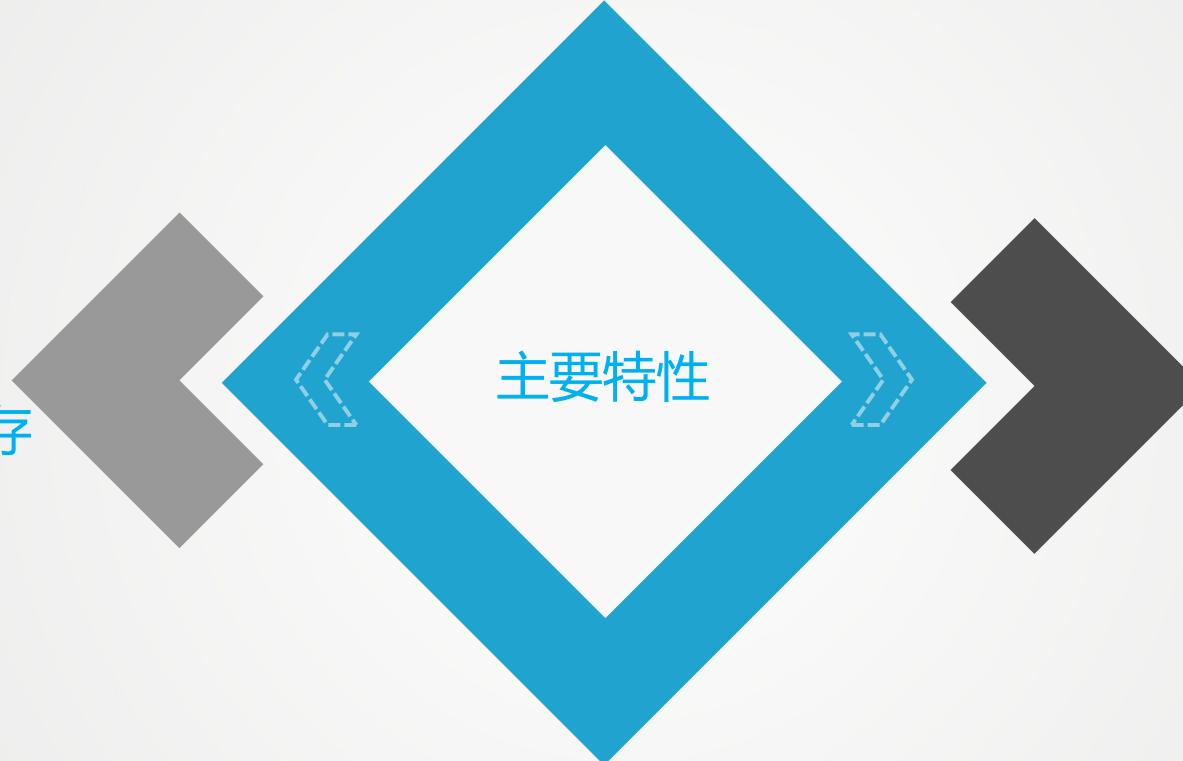
---

随着网络飞速发展，社交网站的普及率越来越高。越来越多的人选择通过社交网站来交友，也就导致源源不断的需求出现，一次又一次地冲击着网站开发者与维护者的技术能力。





- ( 1 ) 增长性
- ( 2 ) 择优链接与随机连接并存



- ( 3 ) 聚合性
- ( 4 ) 关系传递



给“社交网站”定义的关键词是Web2.0和互动。Web2.0是通过在线服务平台让人与人之间交谈，分享经验和参与在线活动



### (1) 商业价值

社交网络平台可以利用用户数据挖掘真实需求从而产生巨大的商业价值。



### (2) 信任

Sinha的研究表明用户更加信任好友所作的推荐,信任在信息推荐中具有重要意义。

### (3) 用户忠诚度

社交网络中的用户忠诚度具有情感的特点, 是一种建立在相互信任关系基础上的忠诚。



( 1 ) 互动模型  
在Bucher对Facebook的Graph Rank算法的研究中,发现算法可以调控用户的行为,以找出最有趣的互动模型,一旦这样的模型被找出,他们就会在Facebook的新闻资讯推荐中出现,因此,随后甚至会有更多的用户以Facebook的算法所预测的方式来进行活动。



( 2 ) 科技传播  
科技传播主要运用社交媒体中的社交网站、微博、微信等平台。社交网站中最有代表性的是泛科学社交网站果壳网。



社交网站逐渐改变了人们的舆论生态和传播行为 ,越来越多的公益组织开始使用社交网站来发布信息 。



### 三、相关技术

---



## 开发工具



( 1 ) 前端  
Html Css JavaScript  
Jquery Bootstrap框架



( 2 ) 数据库  
Mysql



(3)后台  
Python  
Django



(4)服务器  
Mac os  
Django-wsgi

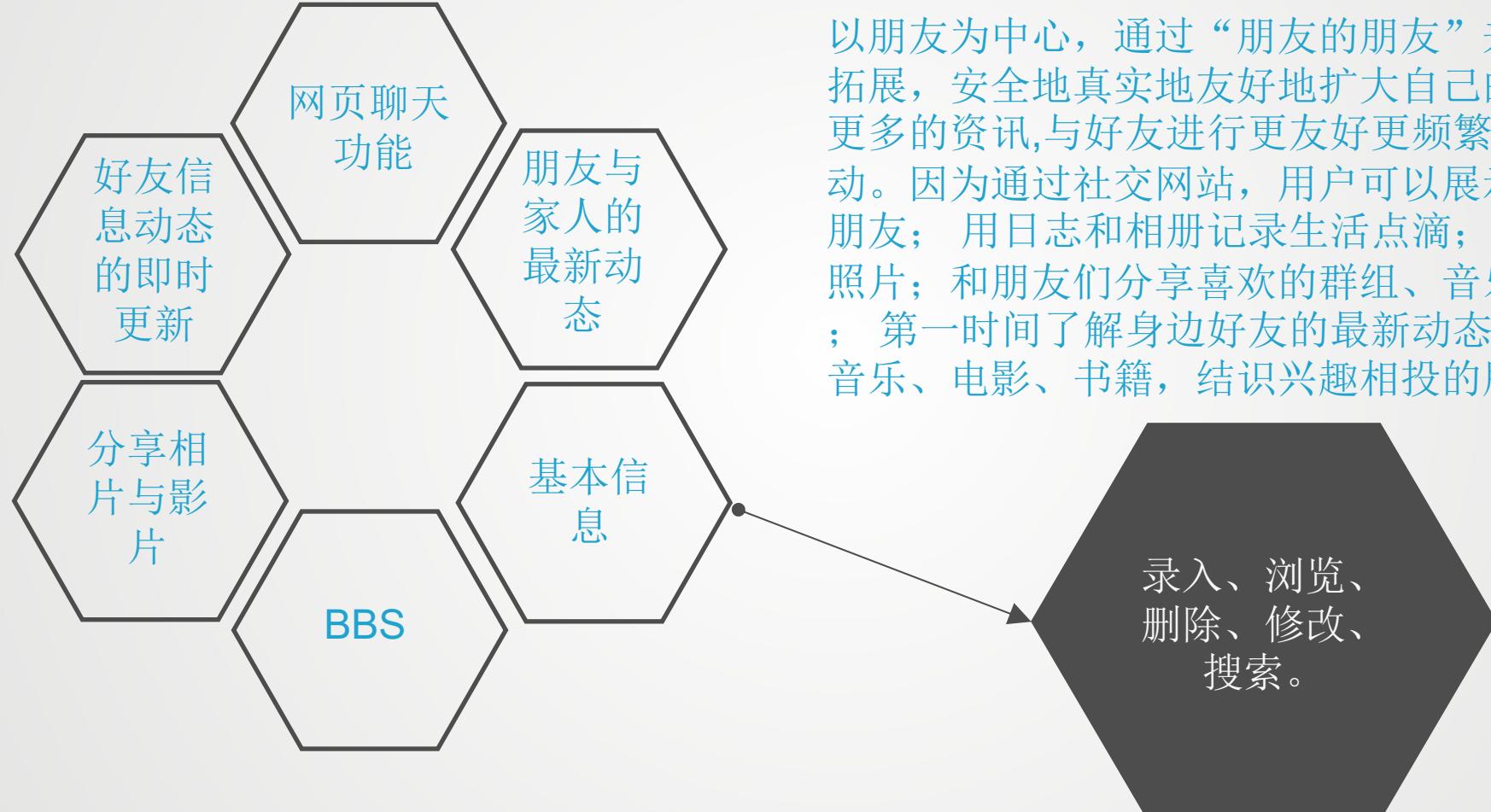


## 四、分析与设计

---

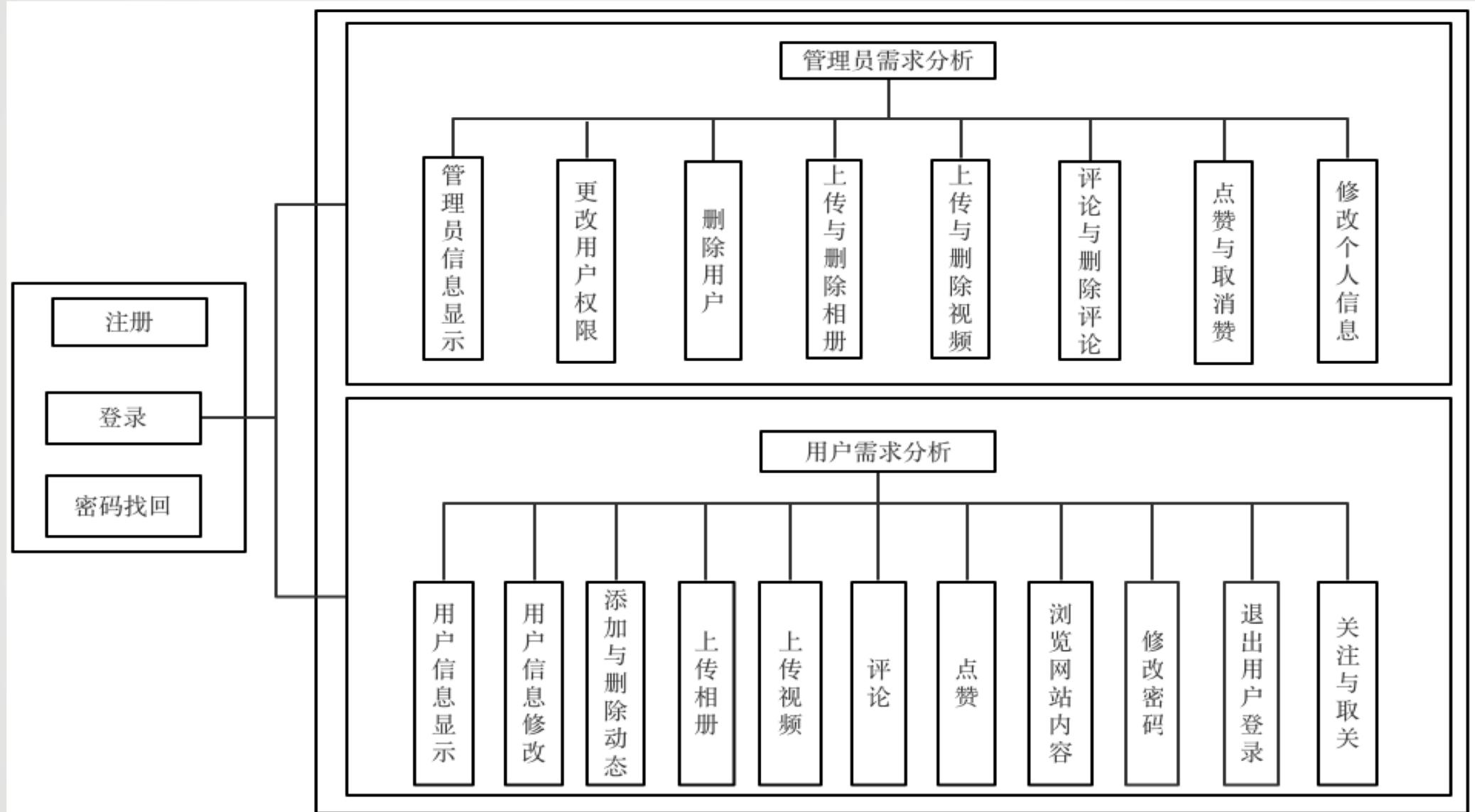


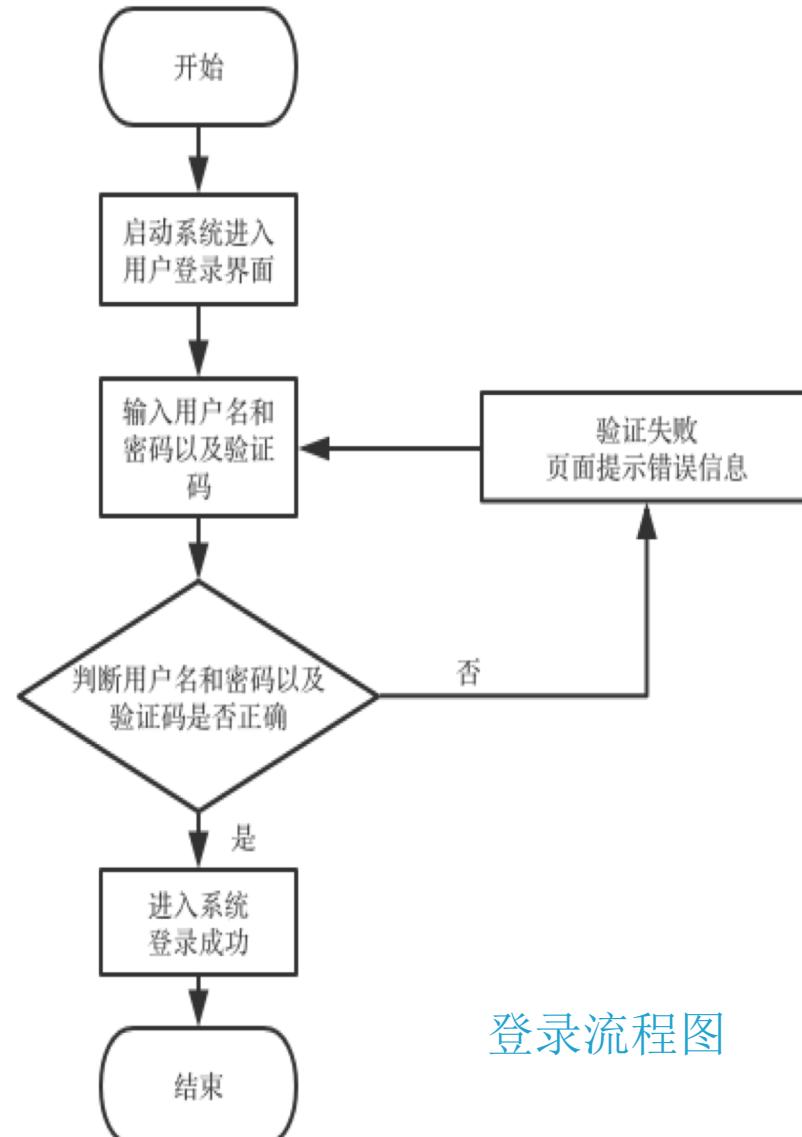
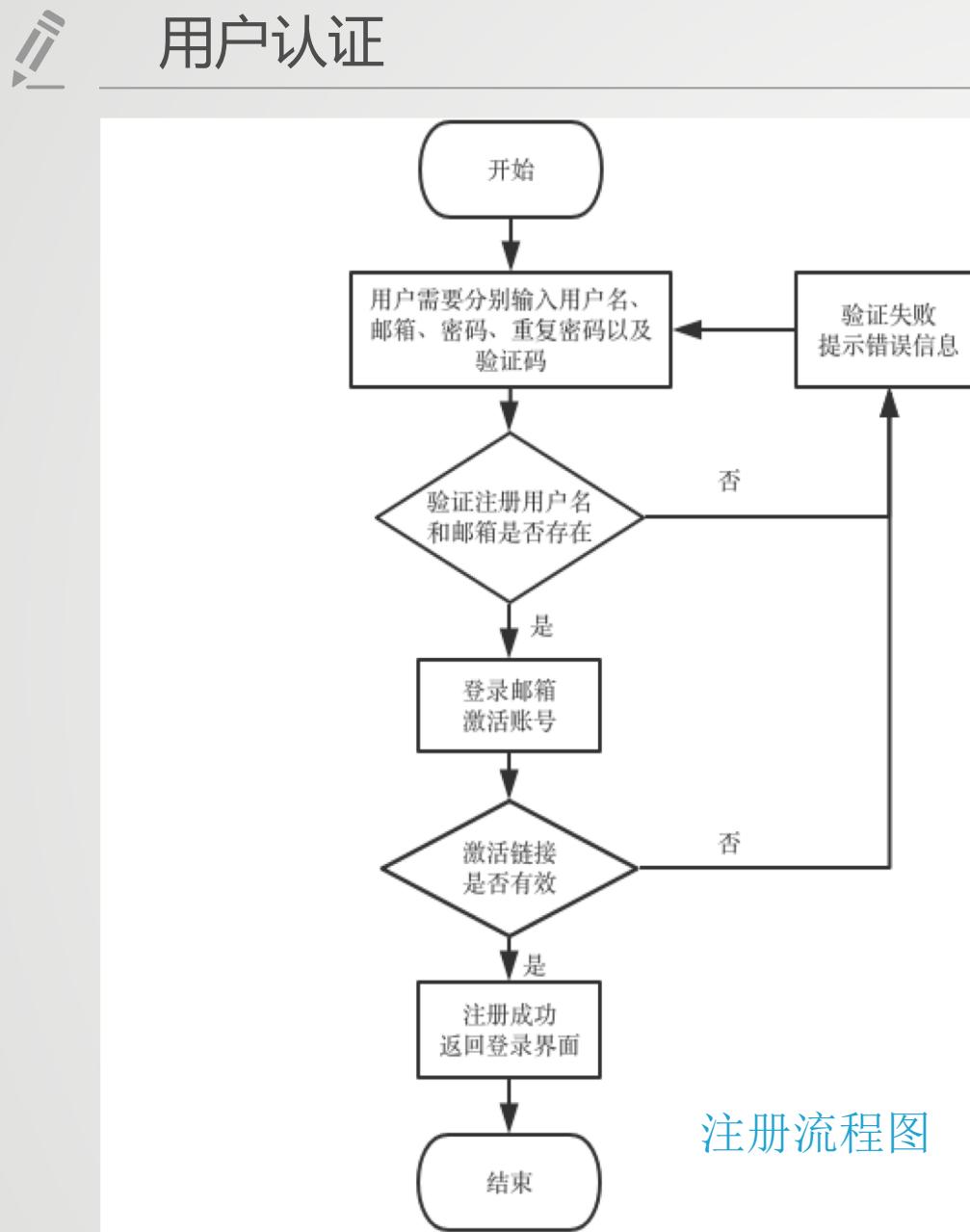
## 相关功能



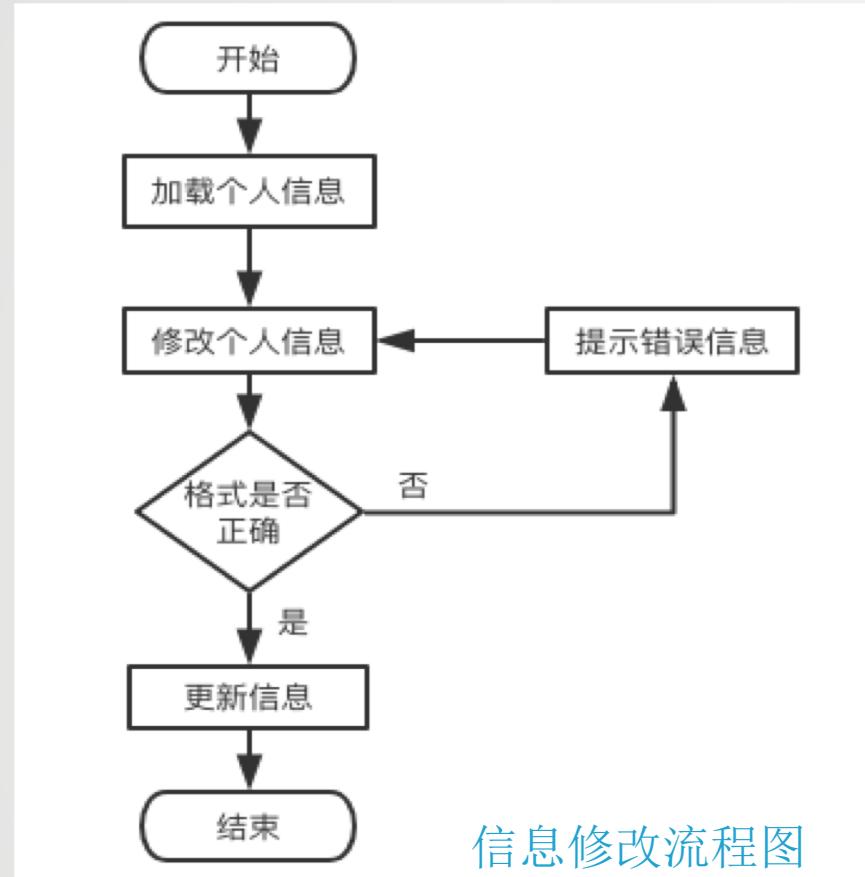


# 功能模块汇总





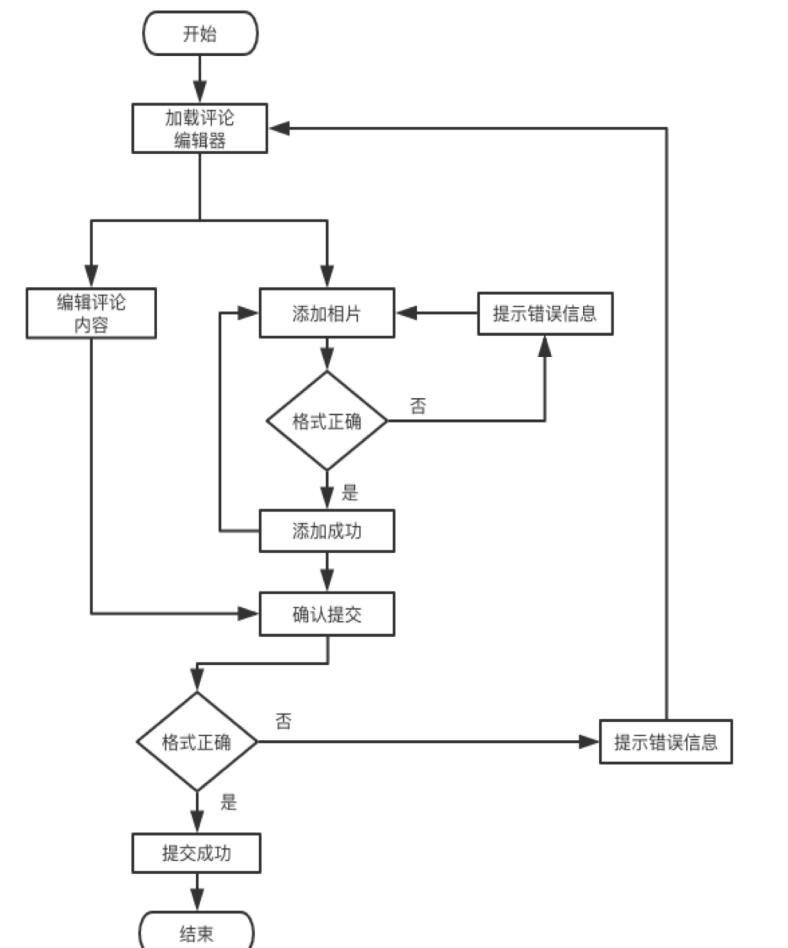
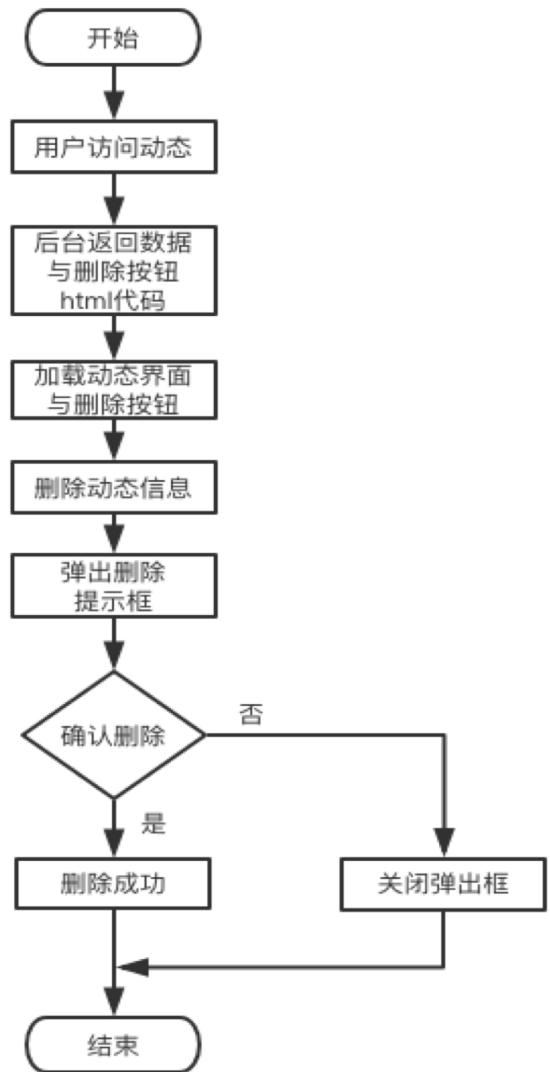
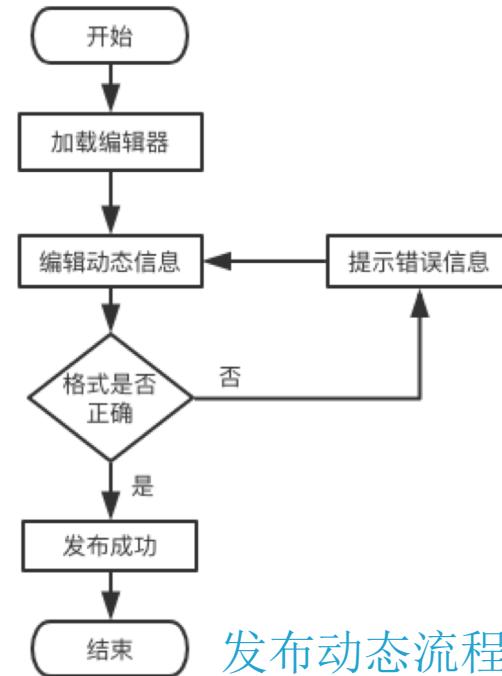
# 用户信息修改



在用户信息管理模块中，用户可以自定义个人主页的内容、包括头像、背景图片、昵称、性别、个性签名、居住地、生日以及工作的修改。



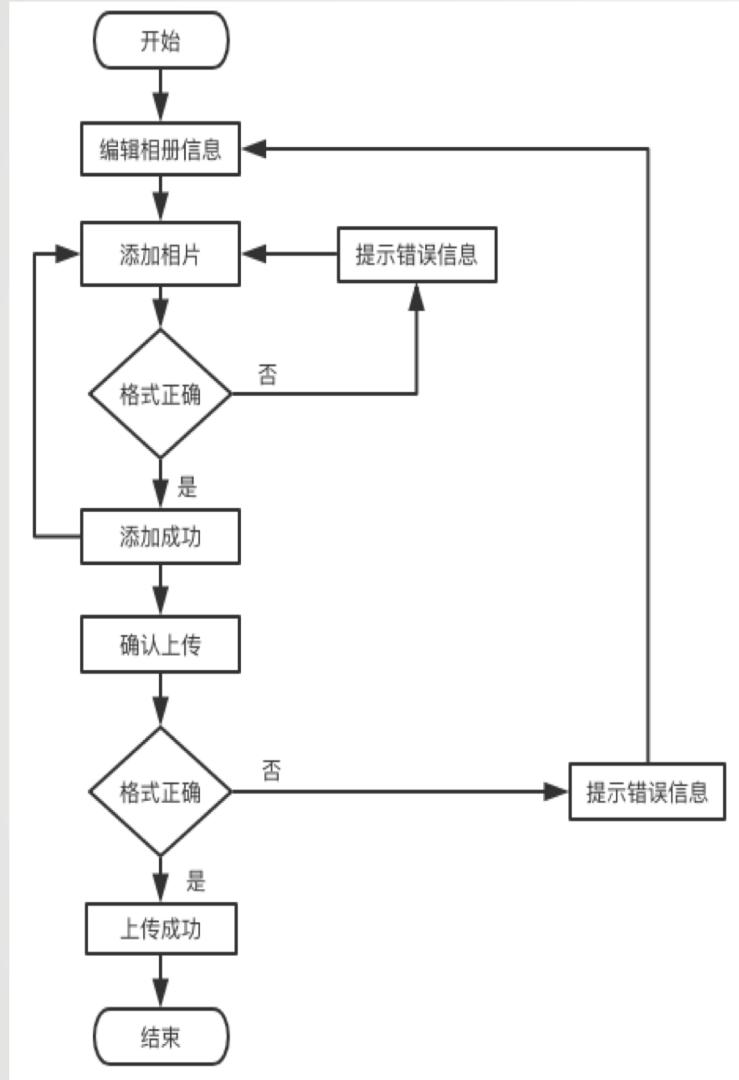
# 动态管理



动态点赞流程图

动态点赞流程图

# 相册与视频上传

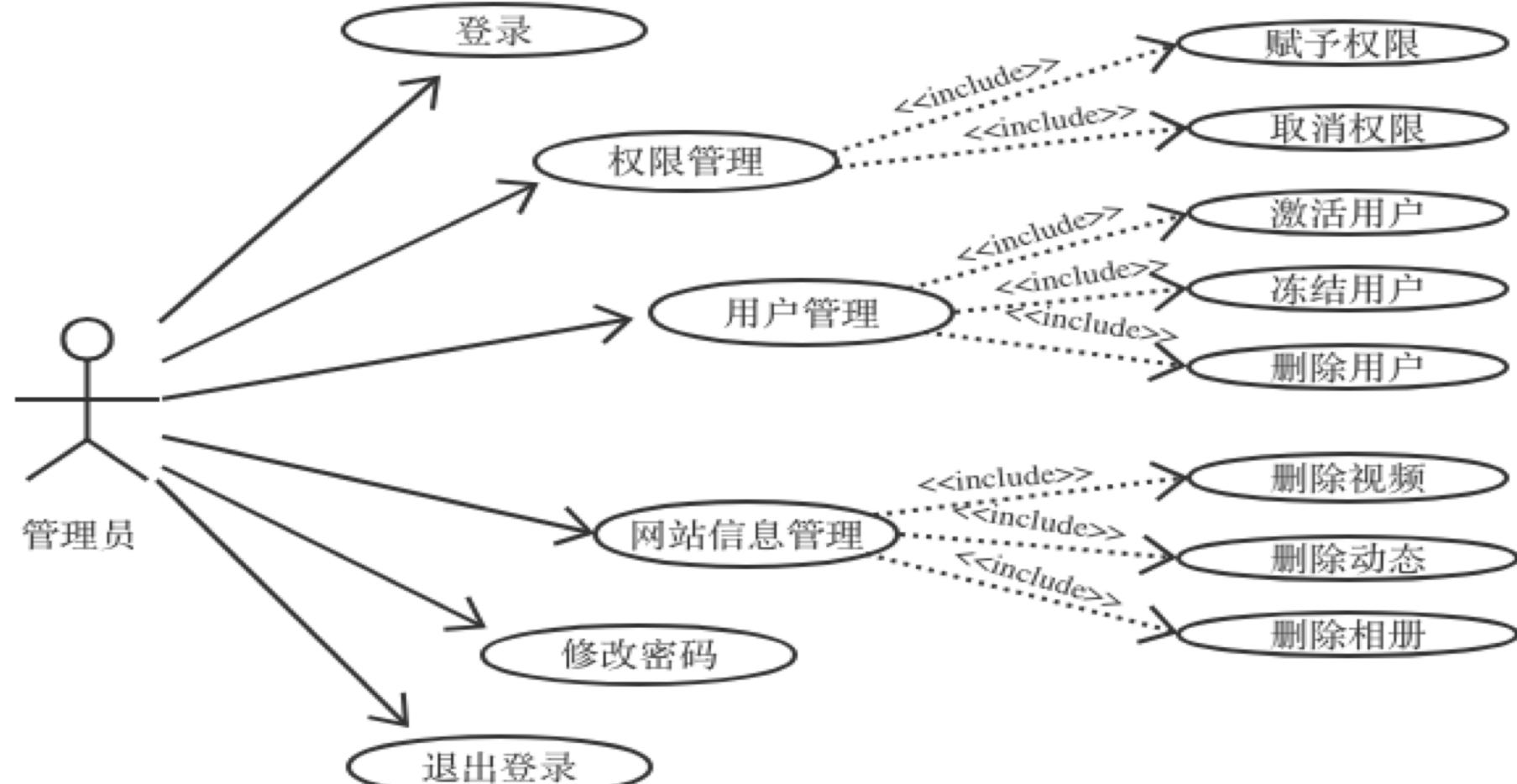


相册上传流程图

视频上传与相册上传功能类似，都是分为两部分。首先上传照片或者视频，然后在确认上传相册或者视频。



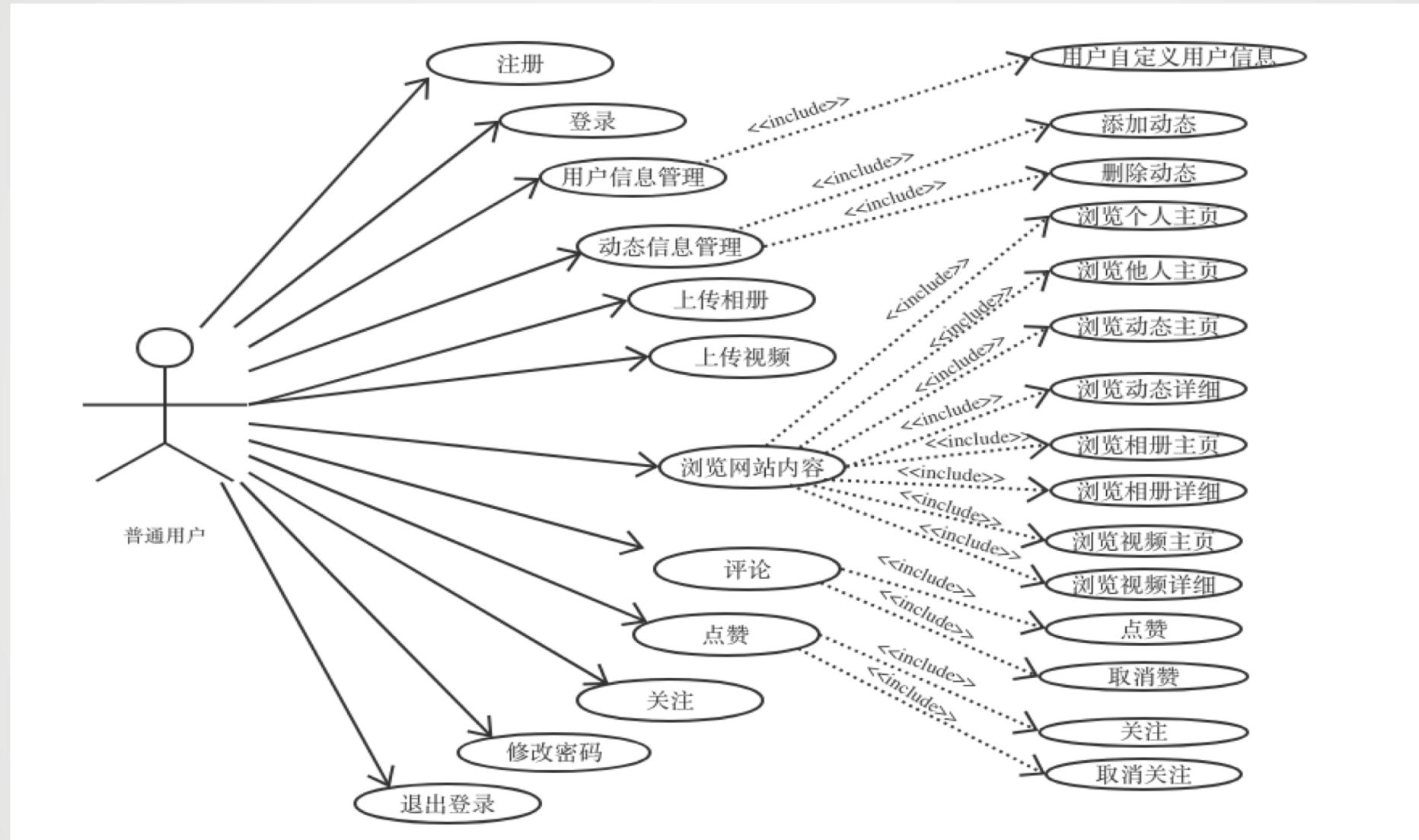
## 管理员用例图



管理员用例图



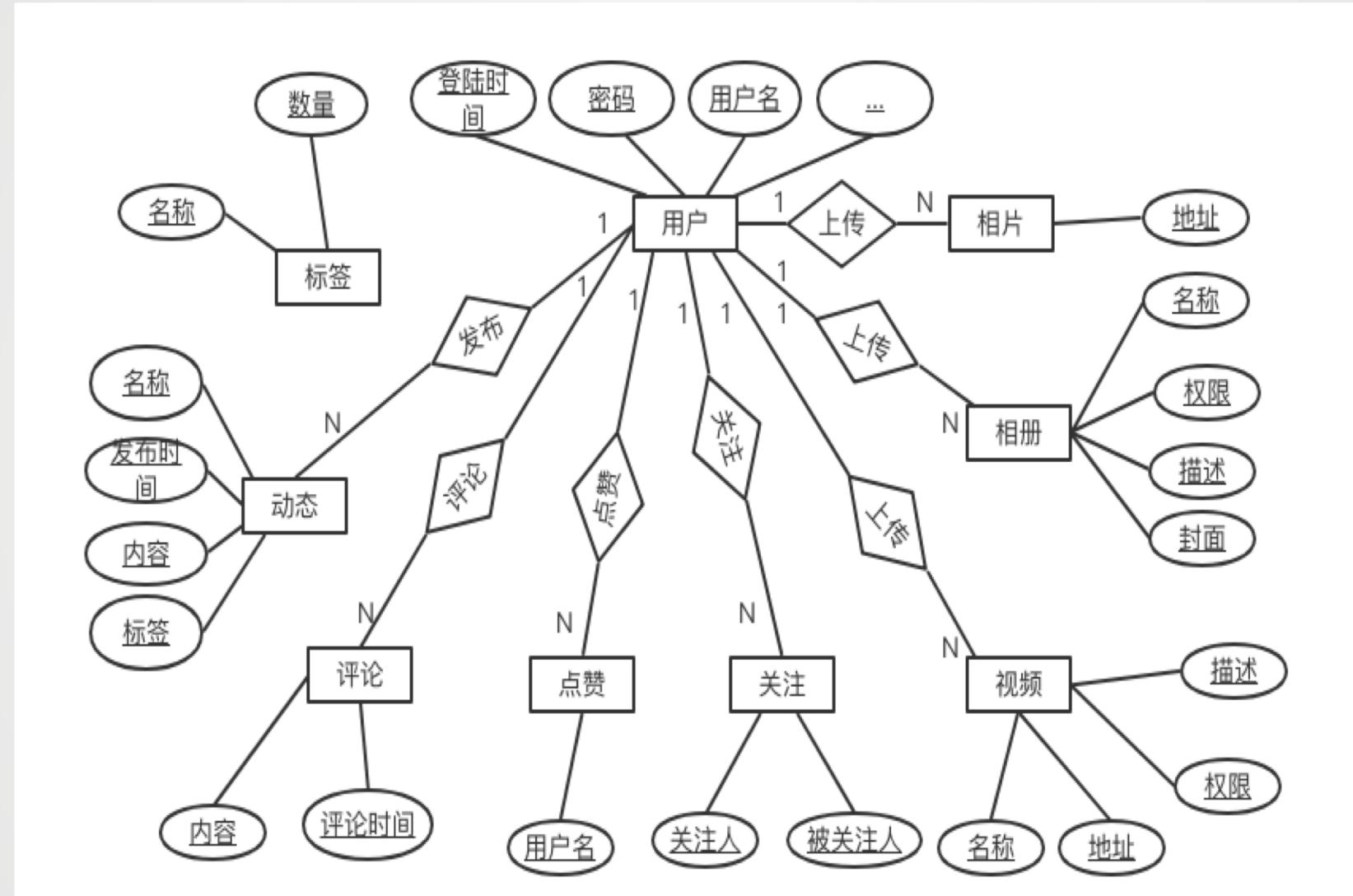
# 用户用例图



用户用例图



# 实体关系图

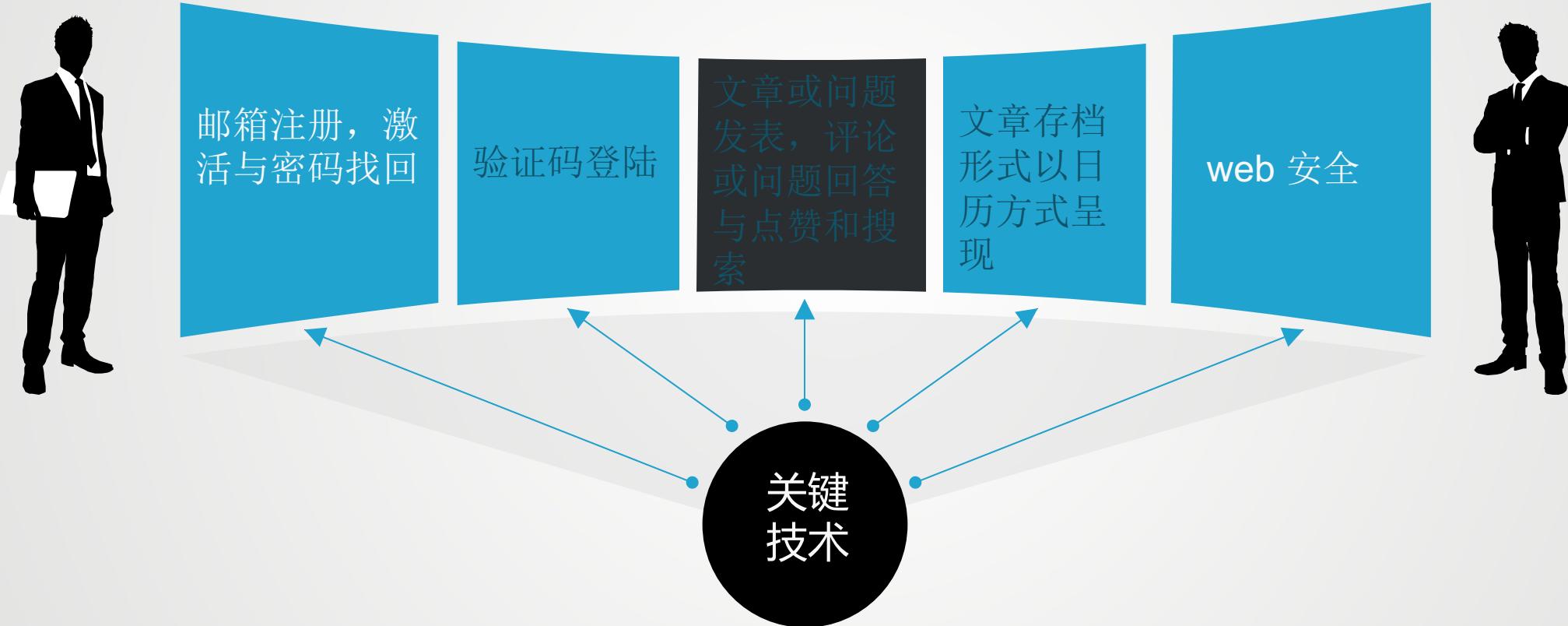


实体关系图



## 五、关键点及其实现

---





## 邮箱注册激活与密码找回



在**settings**配置文件中配置内网邮件服务器，通过邮箱收取验证后台发送的**token**验证码完成用户注册与密码找回功能。

**token**验证码：后台通过脚本产生一串随机码。

注册：将随机码放在激活的**url**中，同时存入到数据库。当用户点击激活按钮时，后台获取该随机码并与存入数据库中随机码进行对比，若相同则置**user.status = true**，否则则激活失败。

密码找回：通过向注册**user**的**email**发送随机码，用户获取验证码后在密码找回界面输入随机码，进行密码找回请求。比对成功后，即可修改密码。



## 验证码登陆

---



验证码登陆：使用**python**的**PIL**模块编写一个可以生成含有随机码的图片的脚本，每当需要使用验证码时，调用一下这个脚本。同时将随机码存入当前**get**请求的**session**当中，当用户输入正确的**username**, **password**和验证码时，后台通过数据库比对**username**的**password**和**session**当中的验证码完成用户登陆，否则登陆失败。



## 文章或问题发表，评论或问题回答

---

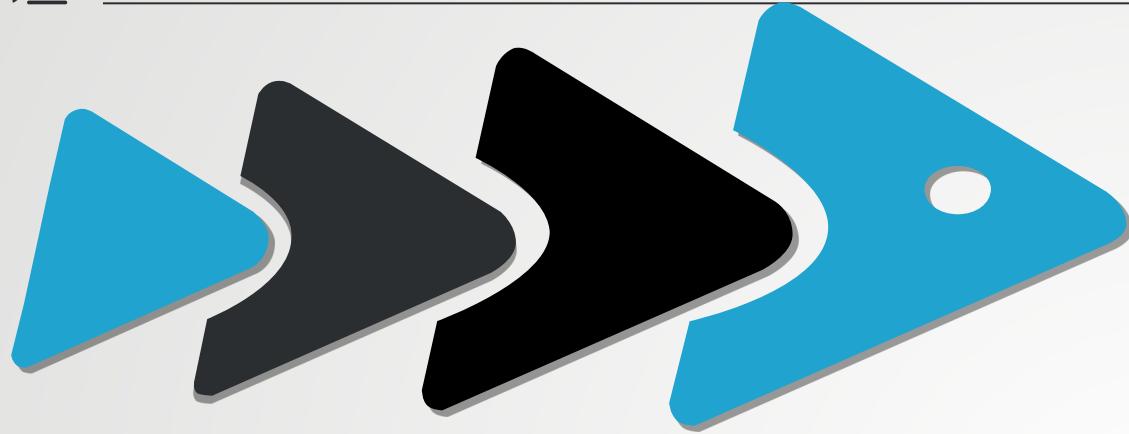


发表：进入编辑界面之后，用户可以输入标题，正文和标签。默认标签是使用空格分开。这里使用**bootstrap**框架进行界面输入框的实现。后台处理过程中，分别有题目,正文和标签三个字段，关键是在处理标签字段时，需要进一步处理，如多个空格，‘/’，回车等特殊字符时要经过过滤后才能存入数据库，否则在前端显示是要出现大规模格式错误。



## 点赞

---



点赞功能：类似于qq空间的点赞，可以点赞和取消赞。

当用户访问某一篇文章时，点击点赞按钮后，通过post请求将当前user的username和文章id传到后台，后台通过文章id获取一张通过外键连接的点赞表中的数据，该表中包括文章ID，status，userid。userid指向user表，文章id指向user表。通过获取点赞表中的数据即可获取该用户是否点赞。如果未点赞，则insert一条数据，如果已经点赞，当再次发出post请求时，delete该条记录。同时通过ajax异步请求控制前端页面是否刷新，即可完成点赞功能。



全文搜索：通过模糊查询的方式如  
`filter(Q(name_icontains='kris') | Q(address_icontains='beijing'))`, q语句时Django中models层的选择查询语句，`contains`即模糊查询。等同于sql语句中的like。当用户输入需要查询的关键字，点击搜索，后台则执行查询语句并返回相应的结果。URL为?keywords=”。

标签搜索：每一篇文章下面都有不同的标签，当用户点击该标签时，即可获取所用包含该标签的文章，也是通过模糊查询的方式。

时间搜索：日历存档。



### 时间搜索：

使用python的**calendar**模块写一个可以产生日历**html**的脚本。获取**html**字符串之后，通过正则表达式获取每一天的数字在字符串的位置，通过**date**字段查询改天是否存在文章，如果存在，将该数字换成一个[标签。\*\*href= ‘ ?date=’\*\*,并且改变数字的\*\*color\*\*属性。最后再将处理过后的日历\*\*html\*\*字符串返回给前端。](#)



验证码二次利用：将产生的图片验证码从保存在文件的方式改为保存在**session**中。

Xss:

字符转义

& -> &amp; ; < -> &lt; ; > -> &gt; ; " -> &quot; ; ' -> &#x27; ; / -> &#x2f;

sql注入：

特殊字符串过滤：

如 “” , or password ‘1’=‘1’这些字符串在与数据库交互之前要经过过滤。

如果用原生的SQL语句需要对查询语句进行预编。如果用Django自带的filter查询语句，则不需要处理。



## 六、成功展示

---



# 感谢聆听

指导教师：王昆  
答辩人：彭磊

院系专业：计算机科学与技术学院  
信息安全

