**嗨社区系统分析与设计**

目录

[1 系统陈述 3](#_Toc359846909)

[2 需求分析 4](#_Toc359846910)

[2.1 用例图 4](#_Toc359846911)

[2.2 术语表 4](#_Toc359846912)

[2.3 活动图 5](#_Toc359846913)

[2.4 用例规约 6](#_Toc359846915)

[2.4.1用例规约 登陆 6](#_Toc359846916)

[2.4.2用例规约 看帖 6](#_Toc359846917)

[2.4.3用例规约 发帖 6](#_Toc359846918)

[2.4.4用例规约 回帖 7](#_Toc359846919)

[2.4.5用例规约 点赞 7](#_Toc359846920)

[3 分析与设计 9](#_Toc359846924)

[3.1 架构分析 9](#_Toc359846925)

[3.2 关键抽象 10](#_Toc359846929)

[3.3 用例实现 11](#_Toc359846930)

[4 用例分析 13](#_Toc359846933)

[4.1 分析类 13](#_Toc359846934)

[4.2 分析类的功能 13](#_Toc359846935)

1 系统陈述：

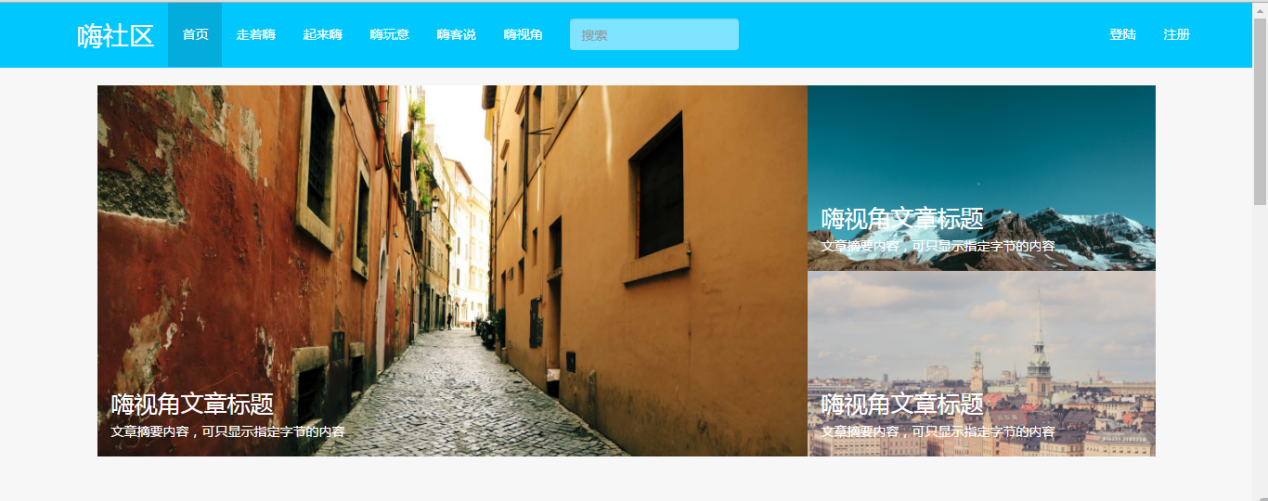
本系统是开发一个在线的社区平台，用户可以在该平台上发布有趣的帖子，可以评论别人的帖子，点赞，回帖，参与活动。可以扩展用户的眼界，该平台包括吃、喝、玩、乐等等小平台，用于用户的交流。嗨社区致力于对用户的衣、食、住、行，吃、喝、玩、乐，做出最好的攻略和交流平台。

2 需求分析：

嗨社区主站主要有以下几个模块：首页、走着嗨、起来嗨、嗨玩意、嗨客说、嗨视角；

嗨社区后台主要用于对用户数据的过滤，处理，收集用户数据。

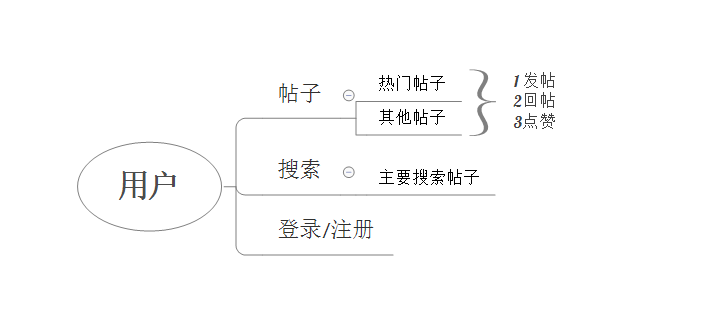
主站：



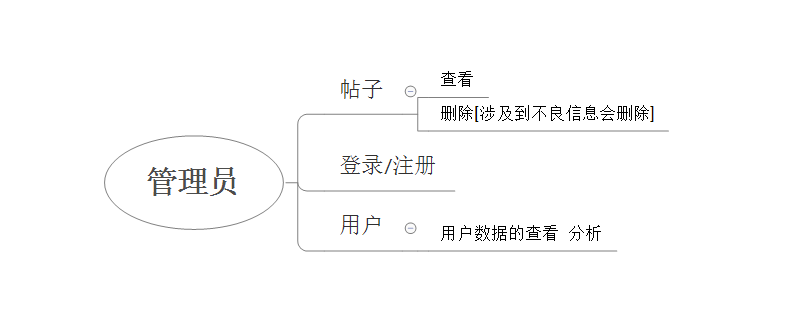
后台：

2.1 嗨社区平台用例图如图2-1所示，主要包括用户登录注册模块，发帖，点赞，回帖，搜索模块等；

主站示例图：



后台示例图：



2.2术语表(HC HiCommunity[嗨社区]缩写)

HC\_UserBaseInfo 用户 [包括admin{有权限控制}]

HC\_NotifyMsgBaseInfo 通知报告表[给用户发的邮件，短信记录表]

HC\_CategoryBaseInfo 模块表 [走着嗨、起来嗨、嗨玩意、嗨客说、嗨视角等模块]

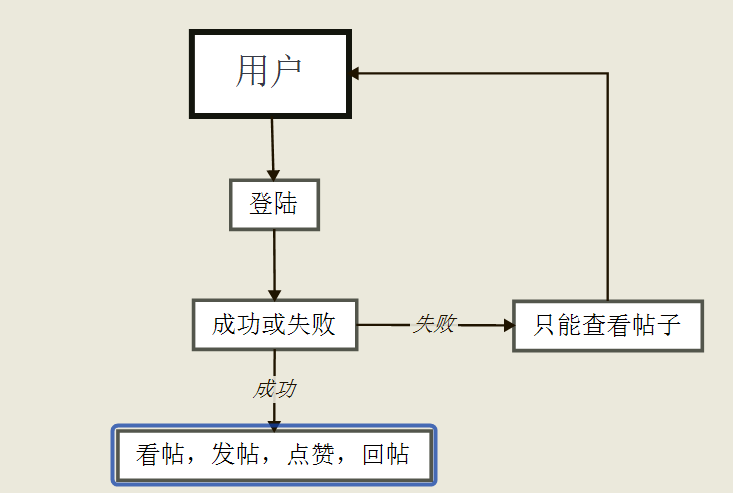
HC\_ThreadBaseInfo 帖子表

HC\_ThreadReplyBaseInfo 帖子回复表

HC\_ThreadTagBaseInfo 帖子标签表

## 2.3 活动图

用户操作流程图：



## 2.4 用例规约

### 2.4.1用例规约 登录

|  |  |
| --- | --- |
| **主要参与者** | **用户及后台管理员** |
| 前置条件 | 输入正确的用户名、密码 |
| 后置条件（成功后的保证） | 登入系统 |
| 基本流程（主要成功场景） | 1. 输入用户名、密码 2. 验证用户名、密码 3. 如验证正确，登入系统 |
| 辅助流程（替代流程） | 输入用户名或密码不正确：   1. 统显示错误信息 2. 提示用户重新输入   系统出现异常 |

### 2.4.2用例规约 看帖

|  |  |
| --- | --- |
| **主要参与者** | **用户或游客** |
| 前置条件 | 用户或游客已经进入嗨社区网站（登陆或未登录都可以） |
| 后置条件（成功后的保证） | 只可以浏览帖子 |
| 基本流程（主要成功场景） | 只可以浏览帖子 |
| 辅助流程（替代流程） | 系统出现异常 |

### 2.4.3用例规约 发帖

|  |  |
| --- | --- |
| **主要参与者** | **用户** |
| 前置条件 | 用户已经登录 |
| 后置条件（成功后的保证） | 能在系统中发帖 |
| 基本流程（主要成功场景） | 1. 用户填写好帖子 2. 用户按下发帖按钮 |
| 辅助流程（替代流程） | 用户输入的帖子不符合规范 及 涉嫌有不良信息，系统出现异常 |

### 2.4.4用例规约 回帖

|  |  |
| --- | --- |
| **主要参与者** | **用户** |
| 前置条件 | 用户已经登录，已经找到能在网站看到的帖子 |
| 后置条件（成功后的保证） | 能在系统中回帖 |
| 基本流程（主要成功场景） | 1. 用户能回复其他用户的帖子（也就是评论帖子） 2. 用户输入回帖的信息 3. 用户按下回帖按钮 |
| 辅助流程（替代流程） | 用户输入的回帖信息不符合规范 及 涉嫌有不良信息，或者网站中没有该帖，系统出现异常 |

### 2.4.5用例规约 点赞

|  |  |
| --- | --- |
| **主要参与者** | **用户** |
| 前置条件 | 用户已经登录，已经找到能在网站看到的帖子 |
| 后置条件（成功后的保证） | 能在系统中对帖子点赞 |
| 基本流程（主要成功场景） | 1. 用户能点赞其他用户的帖子 2. 用户按下点赞按钮 |
| 辅助流程（替代流程） | 网站中没有该帖， 或者 网络出现异常，系统出现异常 |

### 2.4.6 用例规约 后台管理

|  |  |
| --- | --- |
| **主要参与者** | **网站管理员** |
| 前置条件 | **管理员**已经登录 |
| 后置条件（成功后的保证） | 能对系统中的帖子、用户，查看 删除 隐藏 |
| 基本流程（主要成功场景） | 1. 管理员成功登录 2. 对帖子、用户的管理 |
| 辅助流程（替代流程） | 网站中没有该帖、该用户、该管理员，系统出现异常 |

# 3 分析与设计



对于整个项目来说，用了MVC架构，用业务逻辑、数据、界面显示分离的方法组织代码，有利于开发人员的代码阅读，使用了Facade设计模式，建立层次结构系统，降低编译依赖性，除复杂的循环依赖关系，降低了系统的耦合度，使用Facade设计模式，把程序主逻辑放在Facade层，更好的控制了系统的高内聚低耦合。缓存机制采用的是Redis，Redis性能极高 （Redis能支持超过 100K+ 每秒的读写频率），丰富的数据类型（string ， list, set ， hash），丰富的特性 （设置key过期等），支持持久化操作；数据库使用的是关系型数据库Mysql。在存储图片音频视频方面采用了第三方平台七牛云存储。

## 3.1关键抽象

## 本系统的关键抽象包括客户类、帖子类。

3.2 用例实现

用户发帖的用例实现

主要包括 用户类 和 帖子类。

# 4 用例分析

分析类包括：

用户类 、 帖子类 。

## 分析类的功能：

用户： 登陆 发帖 删帖 点赞 浏览 回帖

帖子： 热门帖子 记录所有帖子的信息。