系统设计说明书

项目名称：宠物之家

项目团队：汪星人猫咪联盟

**目录**

[1.引言 3](#_Toc2jzgao)

[1.1编写目的 3](#_Tocez10k6)

[1.2 项目背景 3](#_Tocs1i705)

[1.3 参考资料 3](#_Toc4bm73e)

[2.系统体系结构设计 3](#_Tocfhtx53)

[2.1 体系结构设计 3](#_Tochhss7e)

[2.2 结构层次关系说明 3](#_Tockh2ybe)

[3.系统功能模块 4](#_Tocqdpmos)

[3.1 用户端功能模块 4](#_Tocs2lttf)

[3.2 管理端功能模块 4](#_Tocbw2c59)

[4.用例图、类图设计 4](#_Toc8f51fu)

[4.1 用例图 4](#_Toccolh9f)

[4.2 类图 4](#_Tocu5e4ai)

[5.接口设计 4](#_Tocbxcw69)

[5.1 用户端接口设计 4](#_Toce48jvv)

[5.2 管理端接口设计 4](#_Tocaozz9f)

[6.系统安全与权限设计 4](#_Toccn2a9n)

# 1.引言

## 1.1编写目的

本说明书旨在对“宠物之家”项目系统设计进行说明。主要介绍系统的结构设计、功能模块设计、类图设计、接口设计和系统安全与权限设计。提供系统架构的详细描述，包括系统的组件、它们之间的关系以及它们如何协同工作以满足项目需求。为开发团队提供必要的技术细节，确保开发工作可以按照设计要求顺利进行。

本说明书的读者：系统开发人员、系统测试人员、系统维护人员

## 1.2 项目背景

软件名称：宠物之家

项目开发团队：汪星人猫咪联盟

## 1.3 参考资料

《系统设计说明书》国际规范标准

# **2.系统体系结构设计**

## 2.1 体系结构设计

前端：管理端web，用户端web，

请求：http

控制层：管理端，用户端。接受前端请求，返回请求数据

业务层：管理端：

数据访问层：

数据库：

## 2.2 结构层次关系说明

前端：负责网页界面的显示，获取用户的请求数据并传递给后端。前端分为管理端和用户端两部分。

接口：

控制层：用来接受前端发来的请求和数据，选择合适的业务层进行数据处理并返回给前端数据。

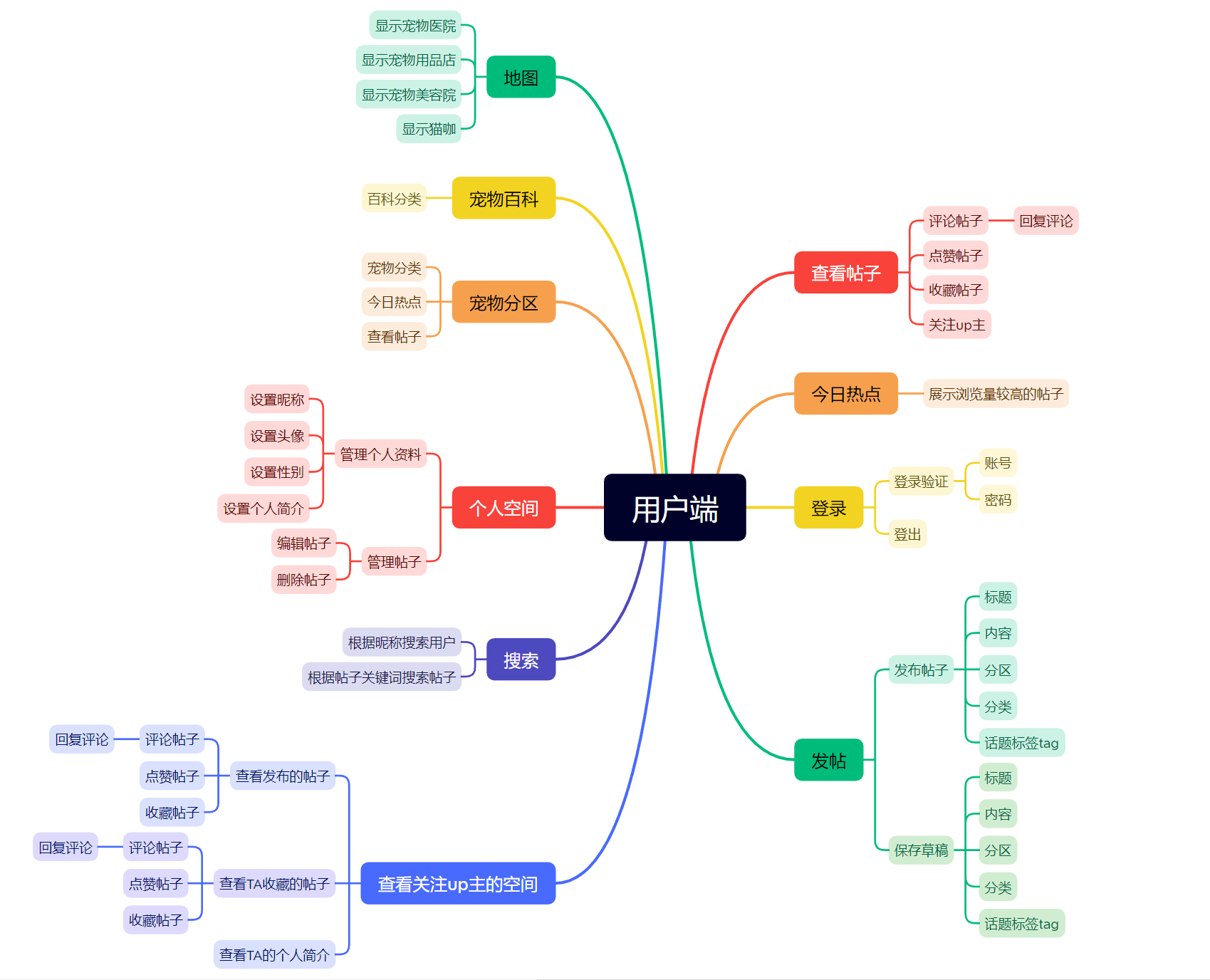
业务层：负责处理由控制层传递来的数据，调用数据访问层的接口查询数据库，将得到的数据进行处理然后返回给控制层。

数据访问层：负责对数据库进行查询、插入、修改、删除操作。

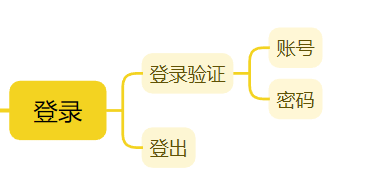
数据库：负责存储网站的所有数据。

# **3.系统功能模块**

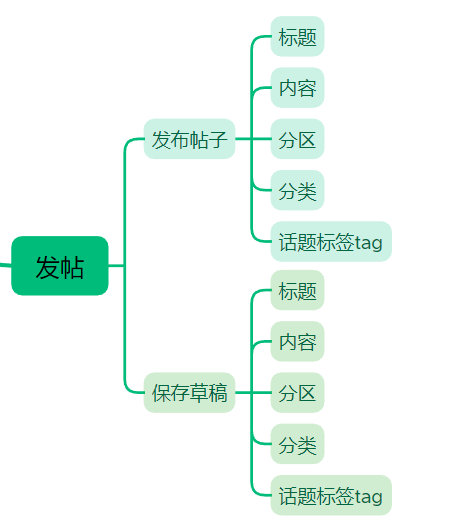
## 3.1 用户端功能模块



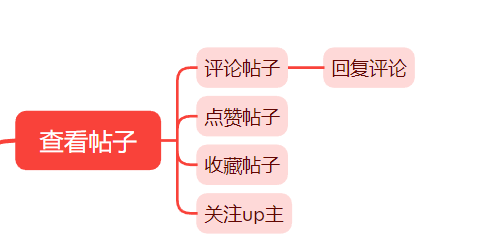
登录：



发帖：



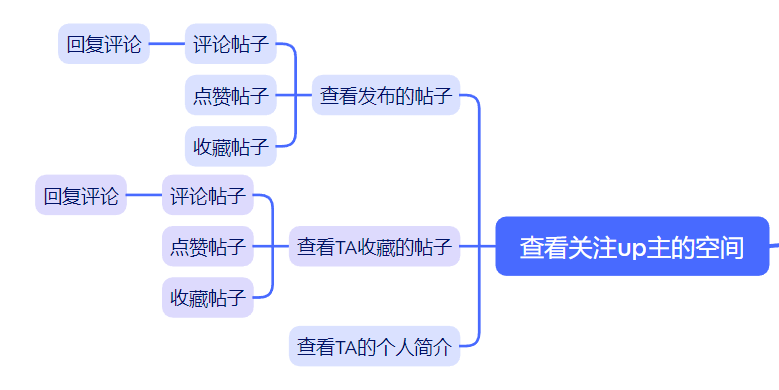
查看帖子：



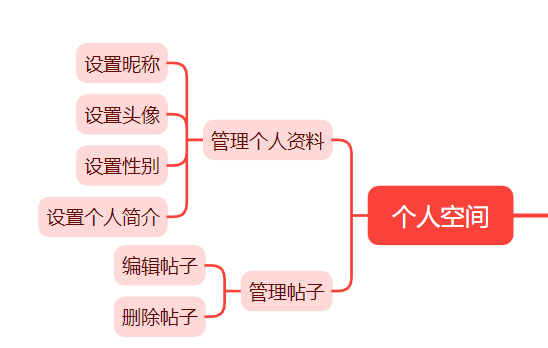
今日热点：



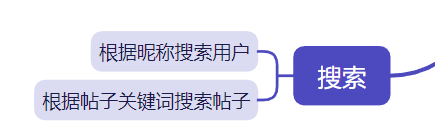
查看关注的up主的空间：



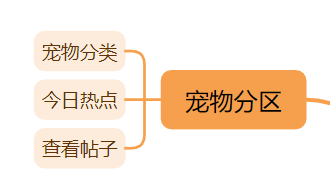
个人空间：



搜索：



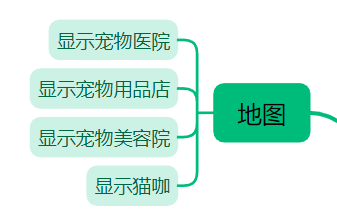
宠物分区：



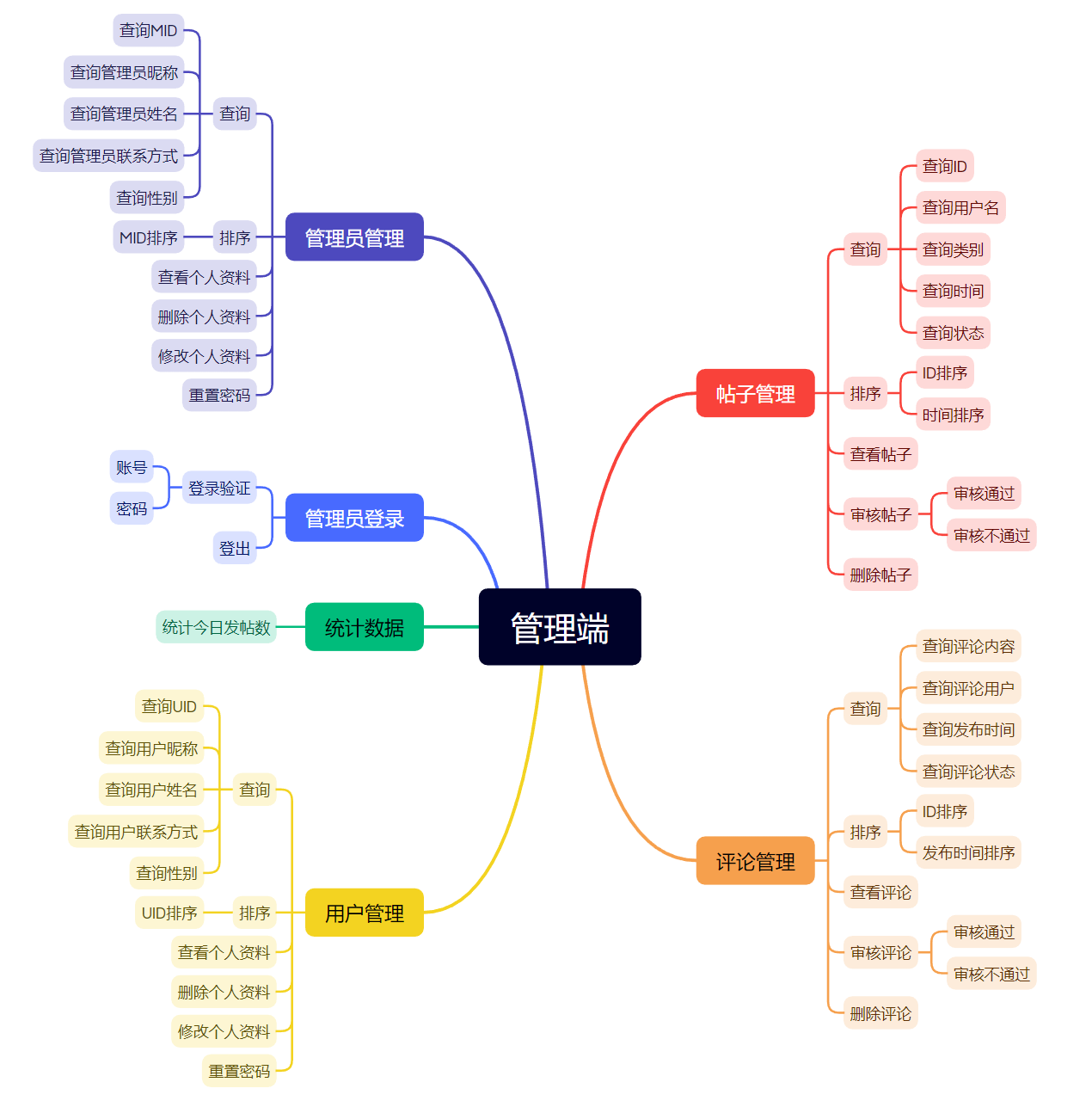
宠物百科：



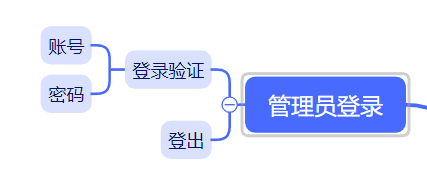
地图：



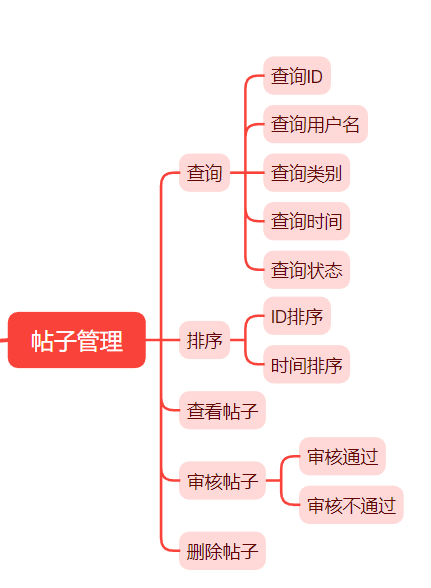
## 3.2 管理端功能模块



管理员登录：



帖子管理：



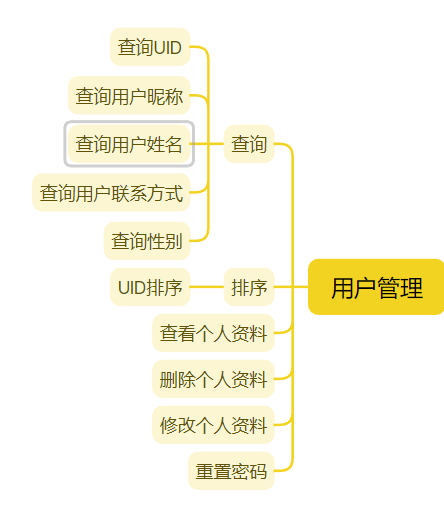
统计数据：



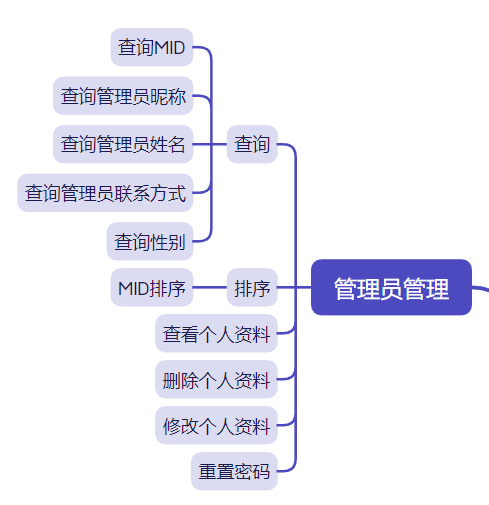
# 评论管理：

# **descript**

# 用户管理：



管理员管理：

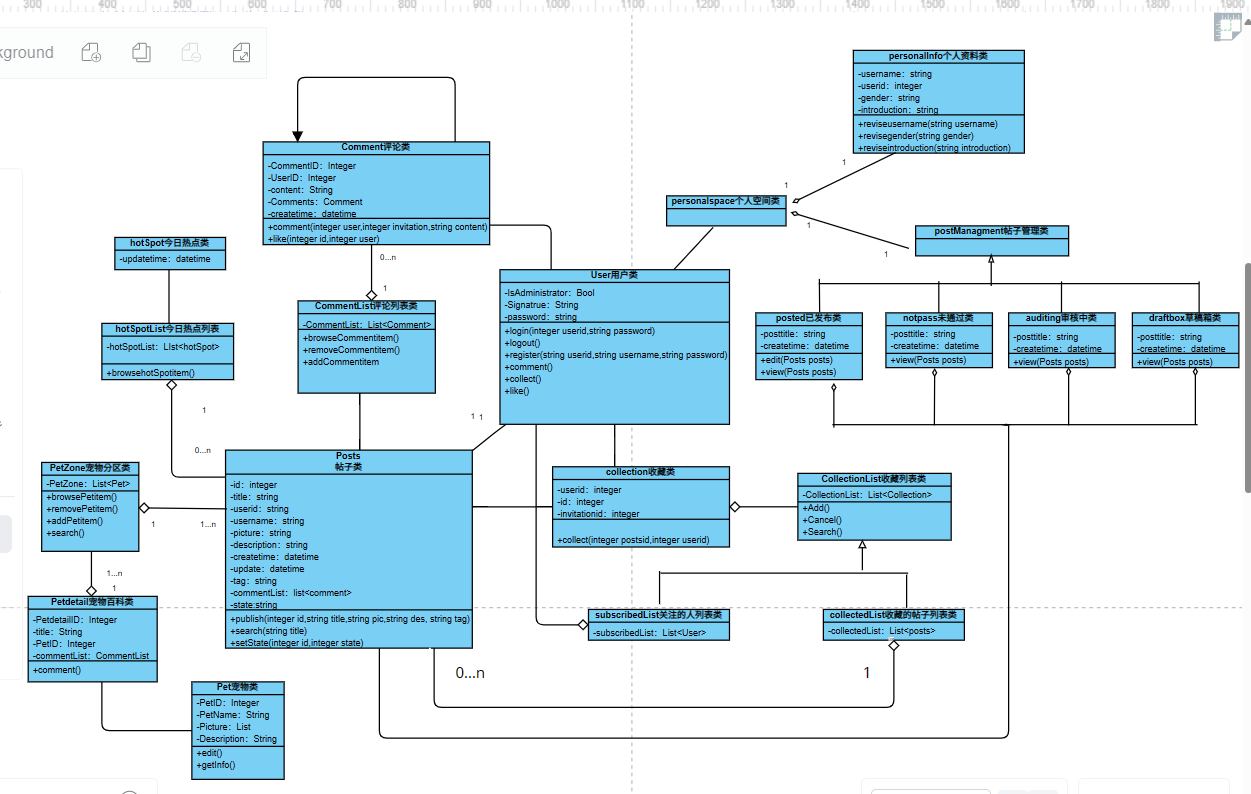


# **4.用例图、类图设计**

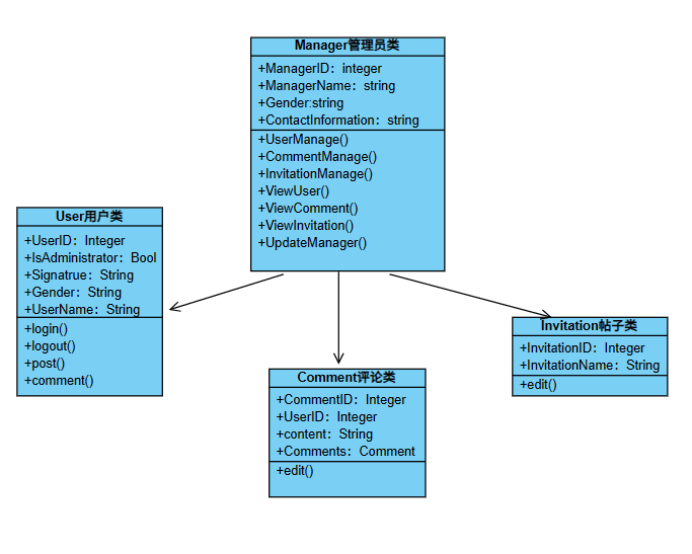
## 4.1 用例图

## 4.2 类图

### 用户端：



### 管理端：



# **5.接口设计**

返回数据封装

{

code

msg

result

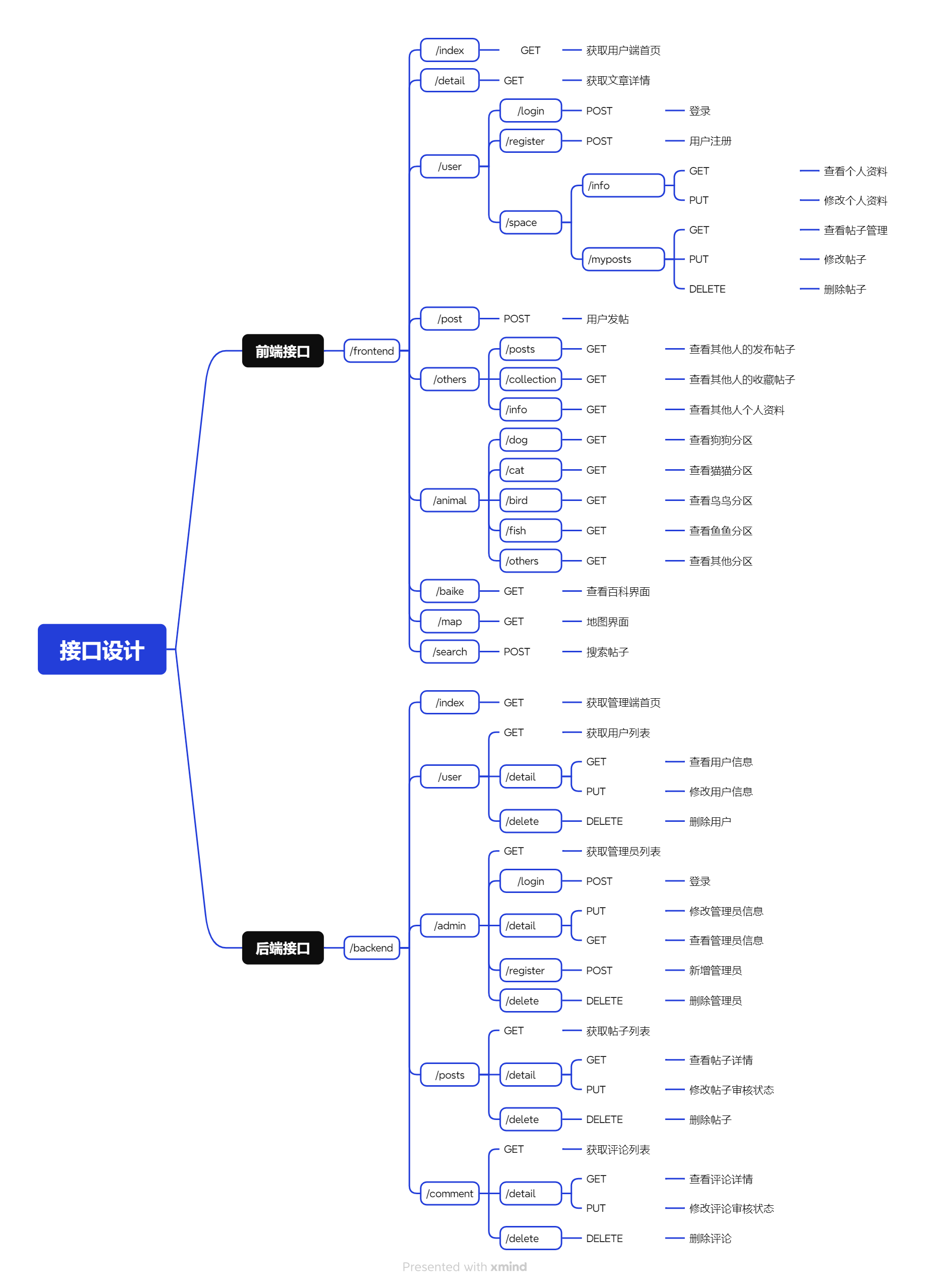
}

用户端

/frontend/xxx

管理端

/backend/xxx



## 5.1 用户端接口设计

用户登录

修改密码

首页

发帖界面

帖子详情

个人空间

关注人空间

宠物专区

宠物百科

地图

## 5.2 管理端接口设计

首页

帖子管理

评论管理

用户管理

查看评论

查看帖子

登录

管理员资料

管理员管理

修改用户

# **6.系统安全与权限设计**

1. **身份认证与访问控制**：
   1. 设计用户身份认证机制，确保只有经过身份验证的用户才能访问系统。
   2. 设计访问控制机制，包括基于角色的访问控制（RBAC）、基于权限的访问控制（ABAC）等，以确保用户只能访问其被授权的资源和功能。
2. **数据加密**：
   1. 对敏感数据进行加密存储，确保即使数据库被非法访问，也无法获取到明文数据。
   2. 在数据传输过程中采用加密协议（如HTTPS）保障数据传输的安全性。
3. **安全审计与监控**：
   1. 设计安全审计日志功能，记录用户的操作行为以及系统的安全事件，用于追踪和分析安全事件。
   2. 配置安全监控系统，实时监测系统的运行状态和安全事件，及时发现并应对安全威胁。
4. **防止常见安全威胁**：
   1. 针对常见的安全威胁（如SQL注入、跨站脚本攻击等）进行相应的防护措施，比如输入验证、输出编码、参数化查询等。
   2. 对系统进行安全漏洞扫描和漏洞修复，及时更新和升级系统组件和依赖库，以确保系统不受已知漏洞的影响。
5. **备份与恢复**：
   1. 设计数据备份与恢复机制，定期对系统数据进行备份，以应对意外数据丢失或损坏的情况，并确保备份数据的安全存储和可靠性恢复。
6. **安全培训与意识提升**：
   1. 组织安全培训和意识提升活动，加强用户和开发人员对系统安全的认识和理解，提高其安全意识和防范能力。
7. **应急响应与灾难恢复**：
   1. 制定应急响应计划，明确安全事件的处理流程和责任人，及时做出响应并采取措施应对安全事件。
   2. 设计灾难恢复方案，确保系统在遭受灾难性事件（如服务器故障、数据中心火灾等）后能够快速恢复运行。