**即时聊天软件：吾亦聊**

详细设计v1.3

**修订历史记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2018/11/6 | V1.0 | 初始化模板 | 杨健威 |
| 2018/12/16 | V1.1 | 补充详细信息 | 杨健威 |
| 2018/12/17 | V1.2 | 完善类图和顺序图 | 张媚，杨玉卿，卢越兴，魏芸 |
| 2018/12/18 | V1.3 | 排版校对 | 魏芸 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目录**

[1 概述 3](#_Toc532903088)

[1.1 系统简述 3](#_Toc532903089)

[1.2 软件设计目标 3](#_Toc532903090)

[1.3 参考资料 3](#_Toc532903091)

[2 术语表 4](#_Toc532903092)

[3 设计概述 6](#_Toc532903093)

[3.1 系统的复用计划 6](#_Toc532903094)

[3.2 系统接口设计 6](#_Toc532903095)

[3.3 对象模型设计 7](#_Toc532903096)

[3.4 系统功能（用例）实现详细设计 9](#_Toc532903097)

[3.5 系统非功能设计 33](#_Toc532903098)

1. 概述
   1. 系统简述

即时聊天软件：吾亦聊，是一款3D聊天软件。用户可以在吾亦聊系统中自由交谈或者私密交流，同时可以体验在日常生活中难以实现的交流方式。

* 1. 软件设计目标

#### 1.2.1 功能性需求

1. 必须实现：
   1. 聊天系统
   2. 用户系统
   3. 交互系统
   4. 管理员系统
   5. 其他修饰性需求
2. 应该实现：
   1. 好友系统
3. 期望实现：
   1. 房间系统

#### 1.2.2 非功能需求

1. 可用性：可用性需要达到99.9%
2. 可修改性：系统内部组件耦合性低，易于修改
3. 效率：系统响应时间在10秒钟以内
4. 安全性：能够放置SQL注入攻击，同时能够抵御一定量的拒绝服务攻击
5. 可测试性：线下测试足够方便
6. 易用性：大多数用户不需要花费超过30分钟的时间来熟悉系统
   1. 参考资料
7. 参考书目：《UML和模式应用》第三版，Craig Larman著
8. 参考书目：《IT项目管理》第7版，Kathy Schwalbe著
9. 参考书目：《软件构架实践》第3版，Len Bass, Paul Clements, Rick Kazman著
10. 术语表

|  |  |
| --- | --- |
| **术语** | **解释** |
| DCD | Domain Class Diagram，显示了模型的静态结构，特别是模型中存在的类、类的内部结构以及它们与其他类的关系。 |
| Web | 网络，本文中泛指计算机网络。 |
| Chat Style | 系统的英文名称。 |
| 房间 | 一个服务端程序可以有多个不同的房间，不同的房间中的模型可互不相同。在房间一中的用户无法看见房间二中的用户，房间二中的用户可以接收到来自房间一中用户的信息仅为私聊频道消息和世界频道消息。 |
| 广场 | 对所有用户开放的多人在线聊天房间。 |
| 家园 | 由用户所私有的多人在线聊天房间。 |
| 虚拟角色 | 用户所控制的3D人物。 |
| 上线/在线 | 表明用户已连接到服务器。与之相对的是“下线/离线”。 |
| 同屏其他用户 | 指可以观察到当前用户的虚拟角色的用户们。 |
| 外观 | 虚拟角色的体型、涂装、装饰等可进行自定义的造型组合。 |
| 涂装/皮肤/预设 | 指虚拟角色3D模型的贴图，可自定义。 |
| 消息 | 聊天的基本单元，由文字、表情图片、其他附件组成。 |
| 转发消息/推送消息 | 指服务器将收到的用户发出的消息，再将消息分发给应当收到此消息的用户们。 |
| 聊天记录 | 可以存储在本地，仅显示上线后收到的消息。 |
| 频道 | 聊天记录的分类，分为“区域”、“世界”、“私聊”、“全部”四个频道。 |
| 区域 | 区域是指一个以用户为中心的几何区域，象征用户的听力范围。当用户在室外时，区域是半径一定的圆形区域，减去建筑截面；当用户在室内时，区域是该层建筑截面和半径一定的圆形的交集。 |
| 区域频道 | 同区域的其他用户发出的消息进入此频道。 |
| 世界频道 | 其他用户发出的世界消息进入此频道。世界消息指有一定限制（如需要使用道具才能发布）、推送到同一服务器服务端程序中所有房间的用户的消息。 |
| 私聊频道 | 与所有其他用户的跨房间私聊往来消息进入此频道。 |
| 全部频道 | 区域、世界、私聊频道的消息全部汇总后的频道。 |
| 动作 | 虚拟角色依照预设的动画做出动作。 |
| 房间物体 | 房间中的可进行互动的3D模型。 |
| 互动 | 执行脚本，改变虚拟角色的动态，或改变房间的样子。 |
| 管理员 | 具有超越普通用户权限的用户，辅助维护社区秩序。 |
| 全服公告 | 管理员发布的特殊消息，具有标识，将转发到同一服务器所有用户处。 |
| 家园主人 | 家园的拥有者。 |
| 访客 | 指目前停留在家园中的其他用户，与家园是否公开无关。 |
| NPC | 指非用户可控角色，在房间中提供某些互动选项。 |

1. 设计概述
   1. 系统的复用计划

库：C#原生库、Javascript原生库、Unity库

框架：管理员端使用Vue作为框架，后端使用Koa作为框架

模式：MVC模式和MVVM模式

构件：

管理员端将DOM组件化；

客户端开发使用Unity引擎预设的组件、Unity Assets Store中的插件。

* 1. 系统接口设计

#### 3.2.1 提供给用户的接口：

* 1. 管理员端Web界面
  2. Unity客户端界面

#### 3.2.2 系统内部的接口：

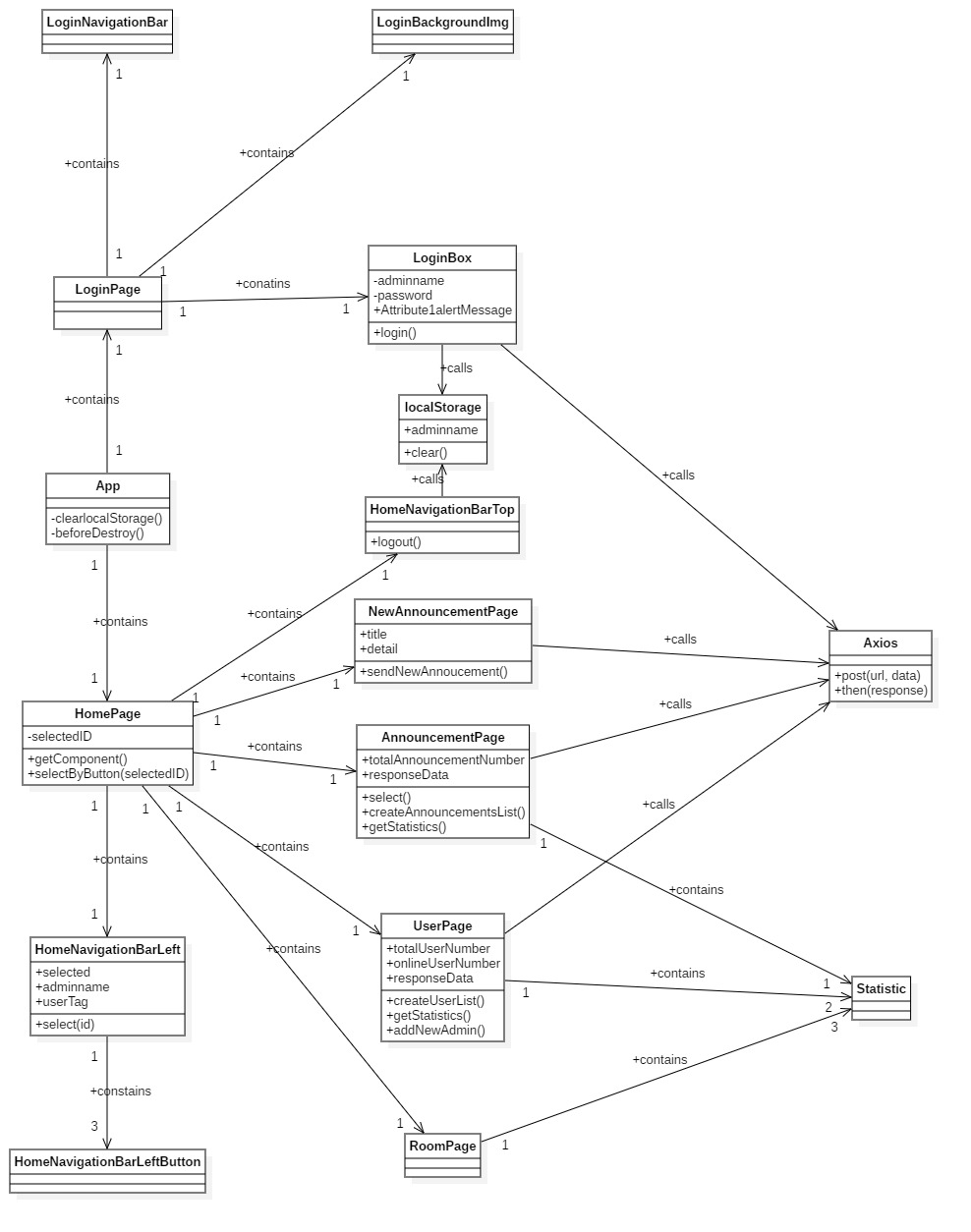
详情见api.html

#### 3.2.3 系统外部的接口：

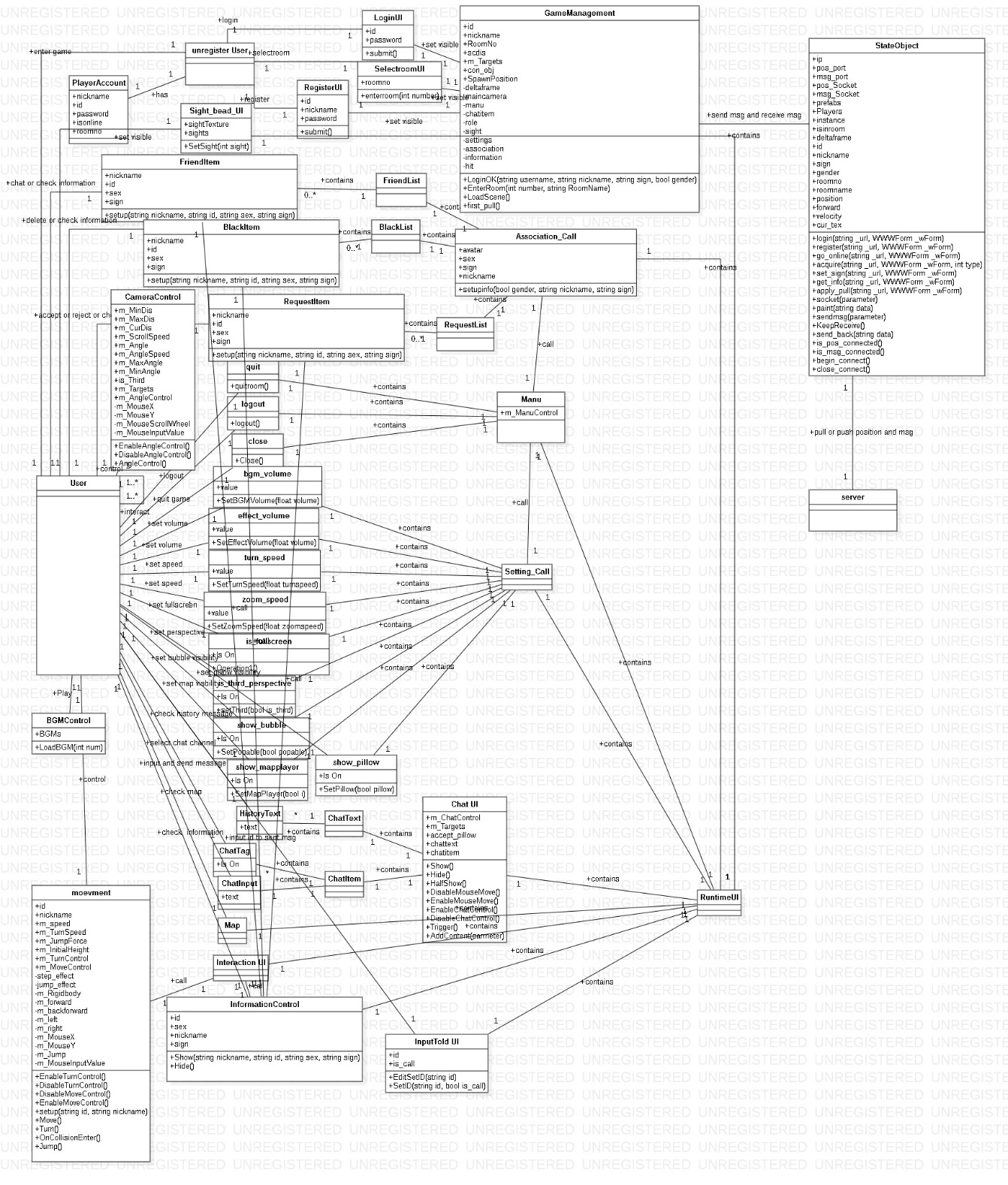
无

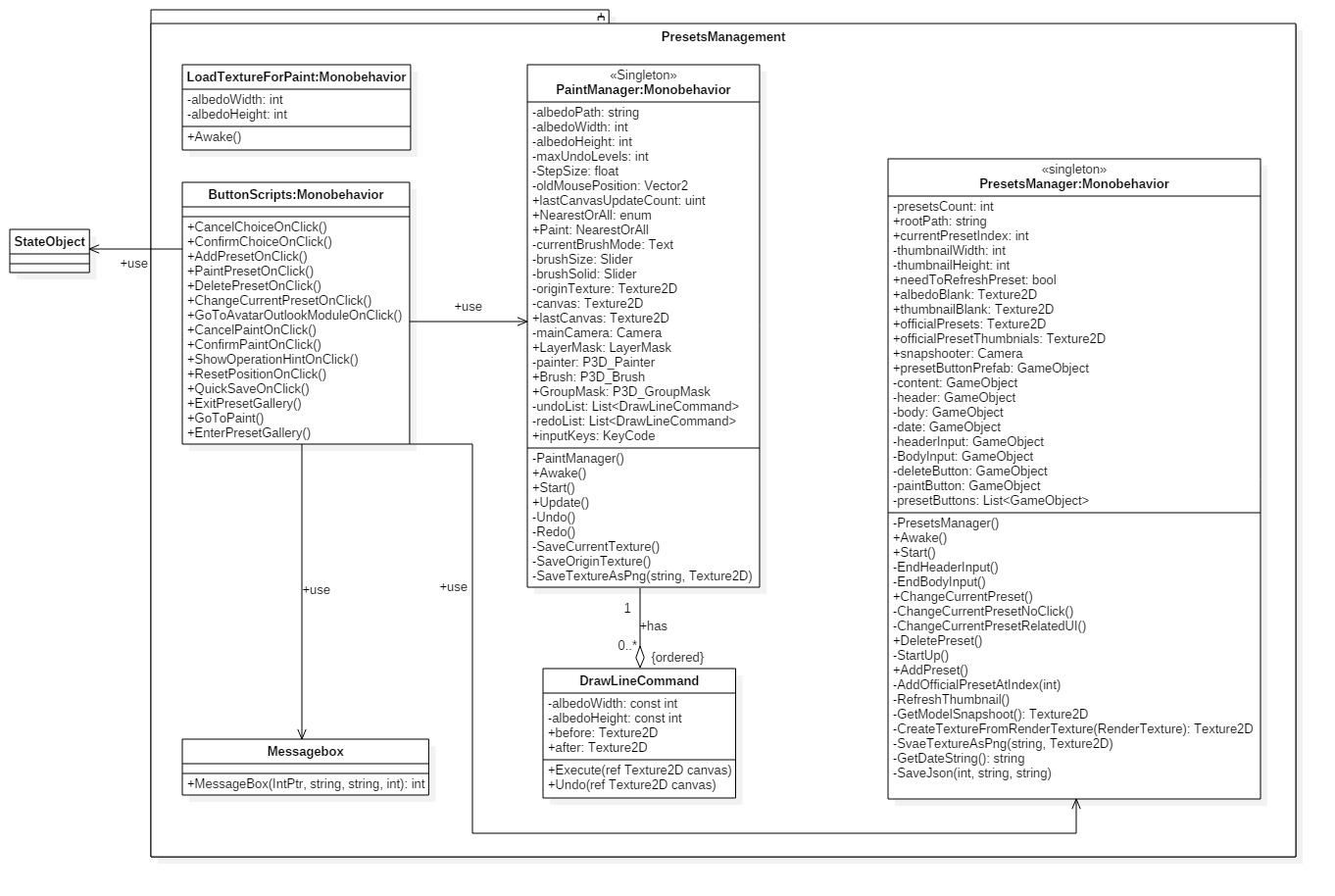
* 1. 对象模型设计

管理员端类图：

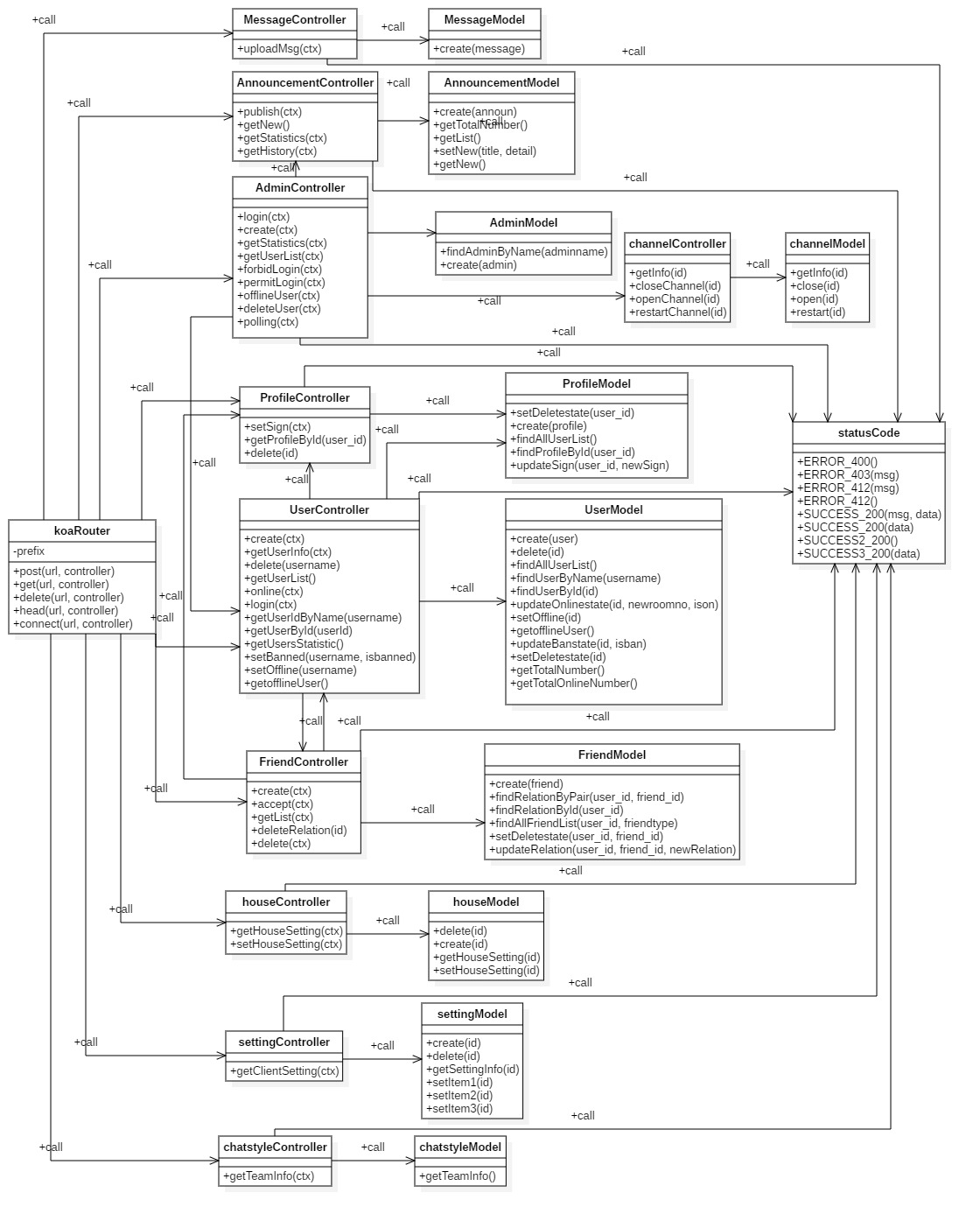


前端类图：





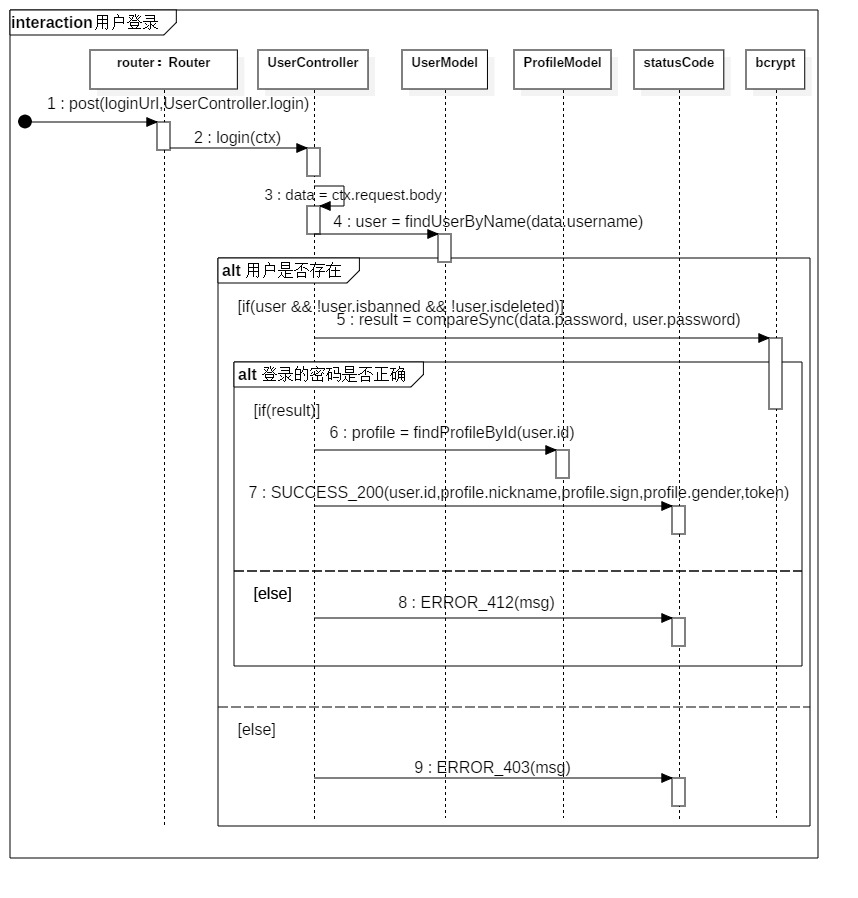
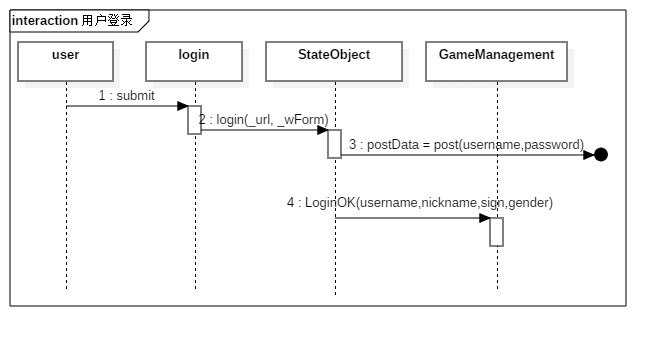
后端类图：



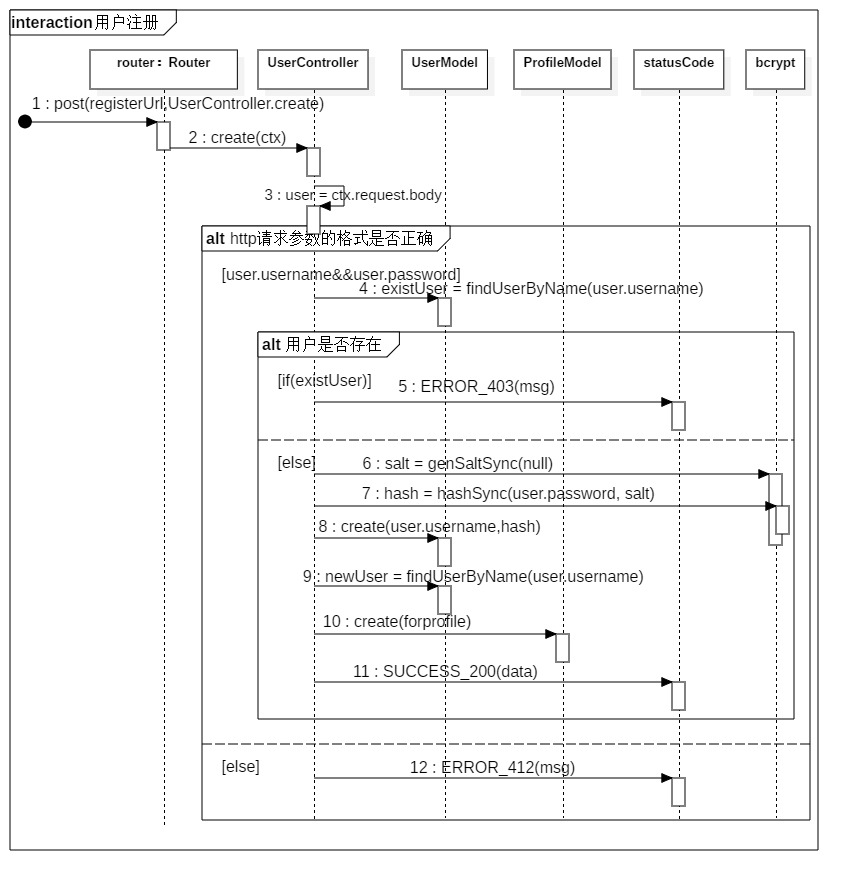
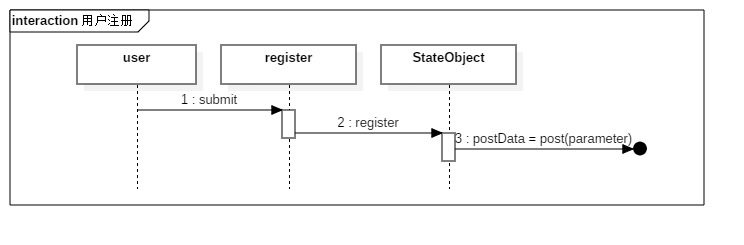
* 1. 系统功能（用例）实现详细设计

#### 3.4.1 系统功能（用例）实现详细设计

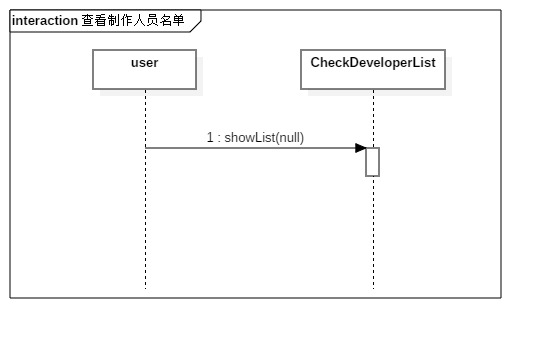
#### 用户登录



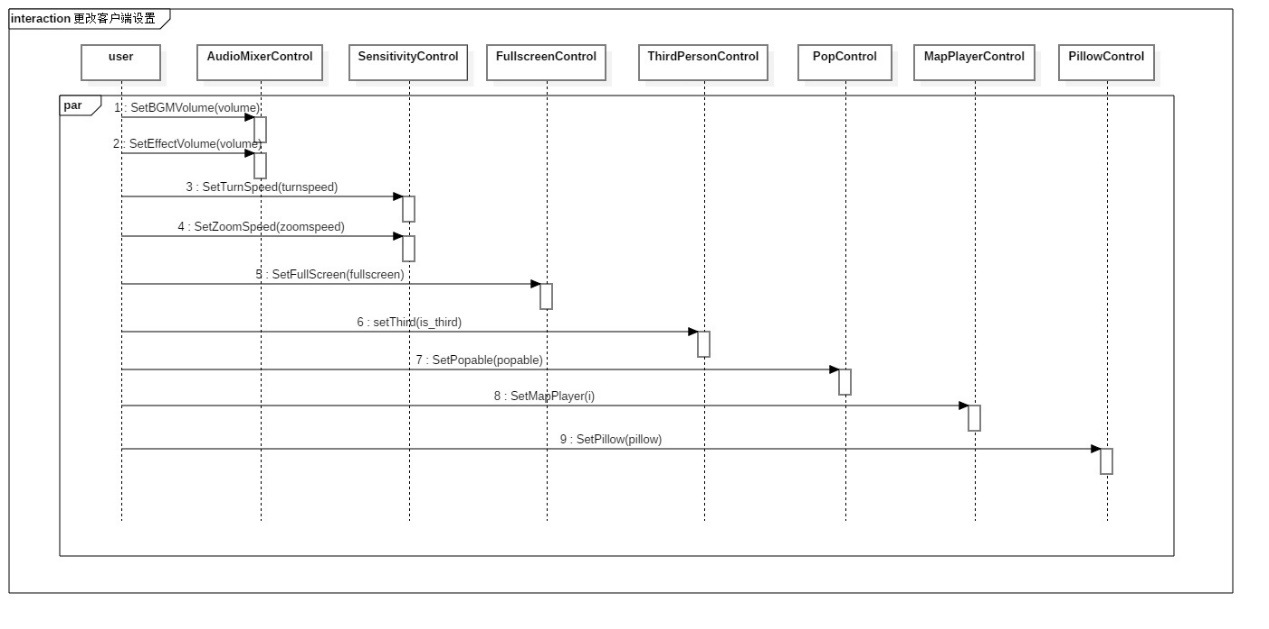
#### 用户注册



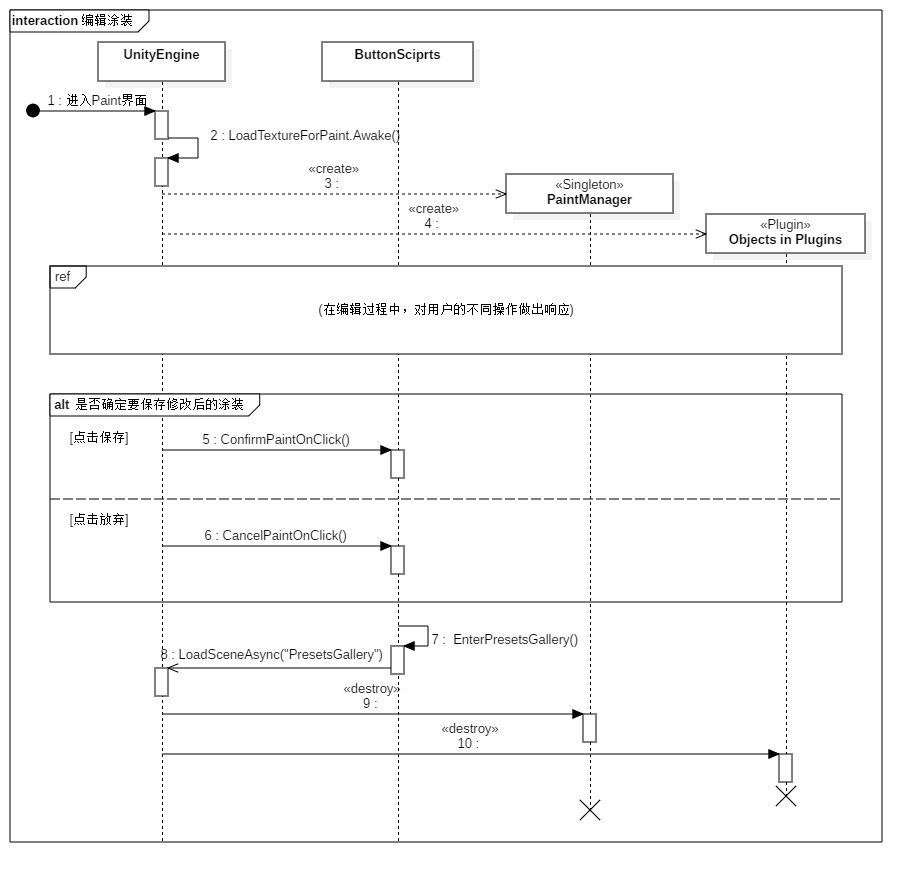
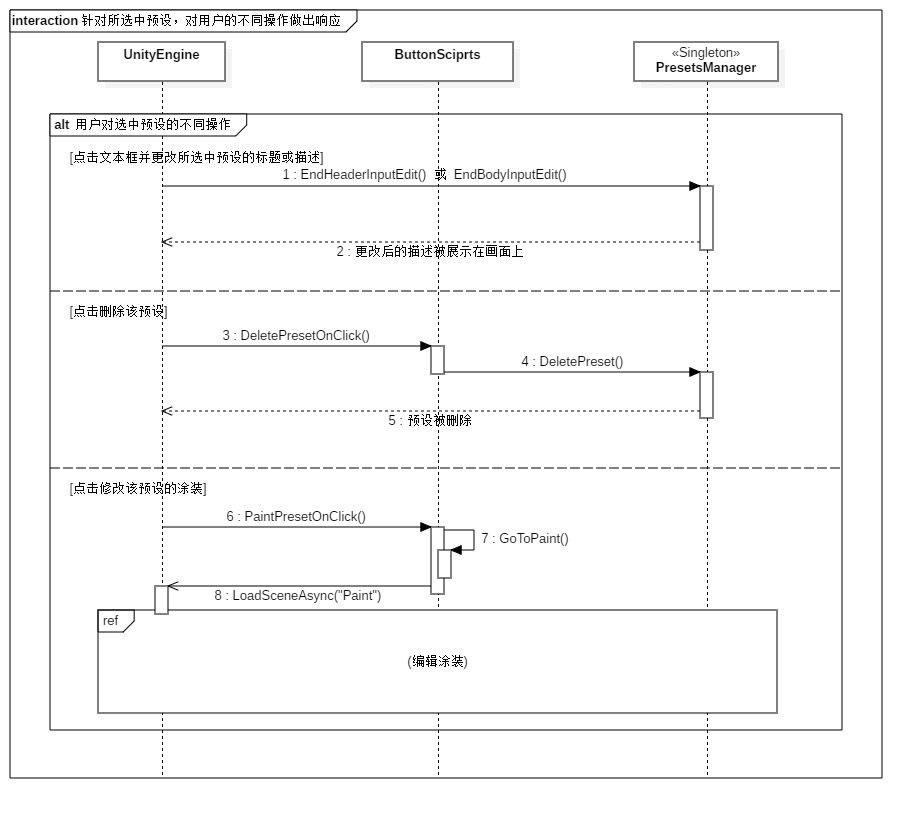
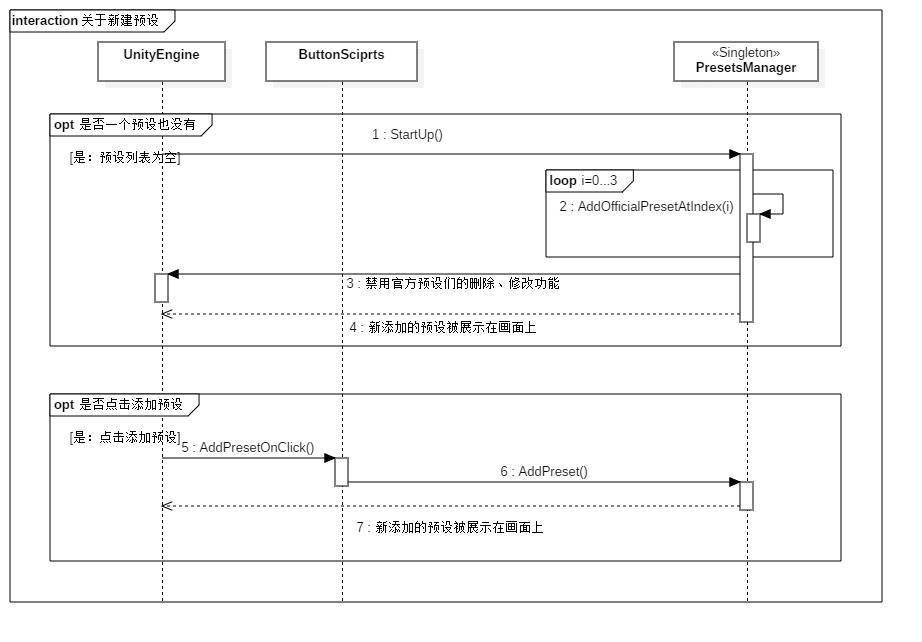
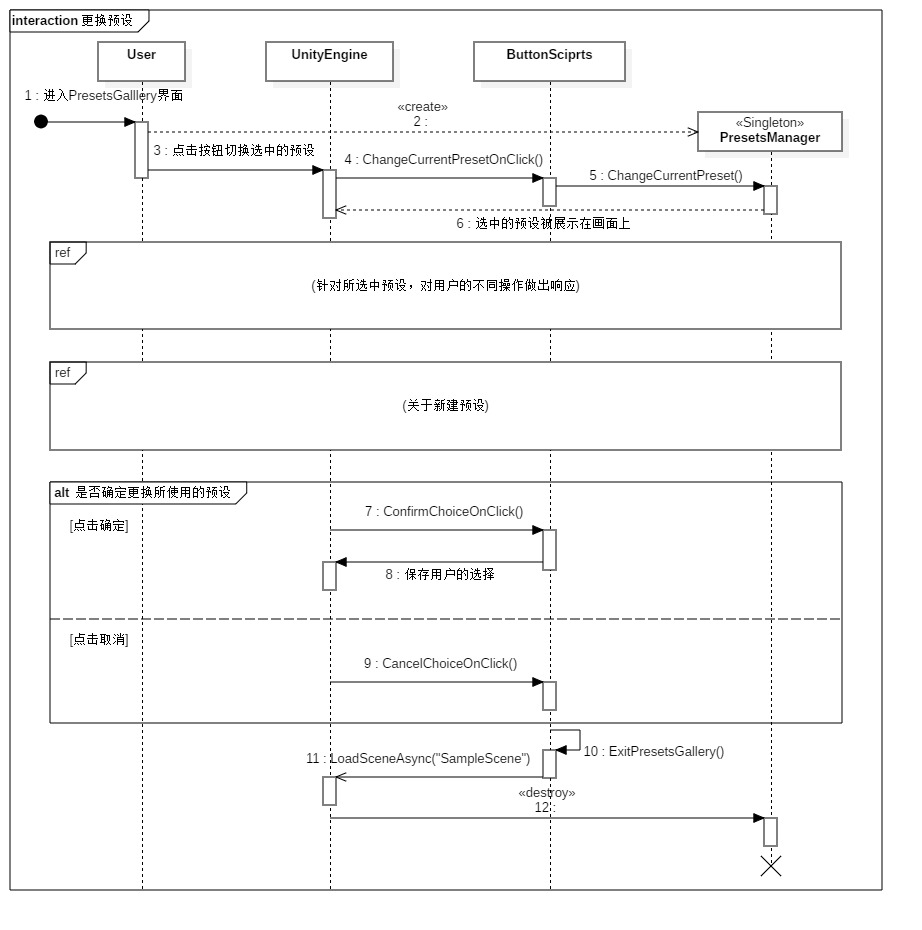
#### 查看制作人员名单



#### 更改客户端设置

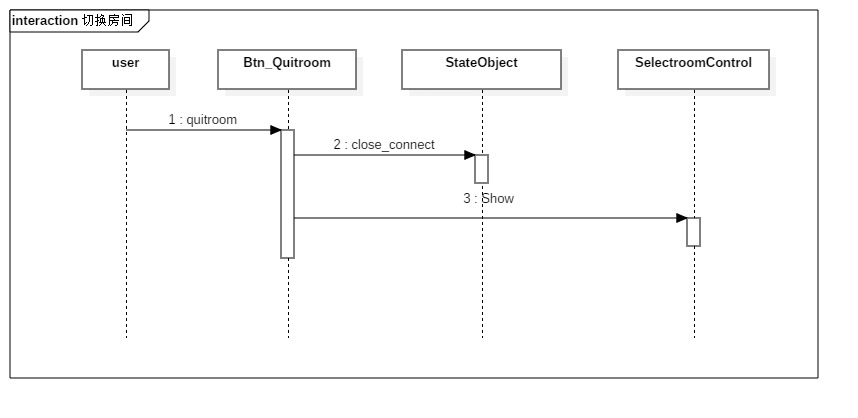


#### 调整虚拟角色外观

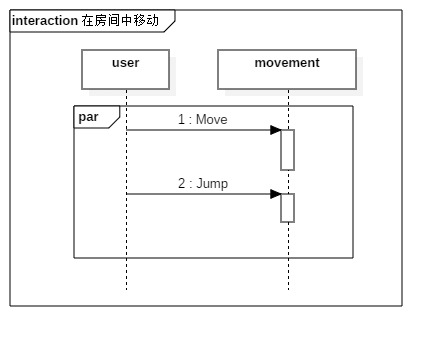




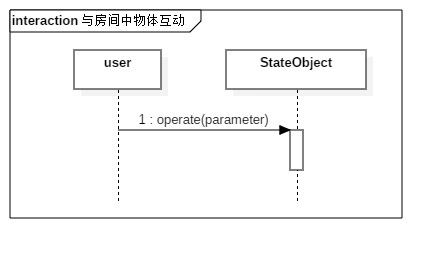
#### 切换房间



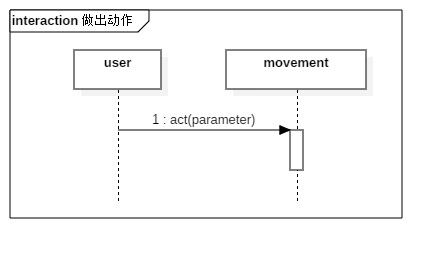
#### 在房间中移动



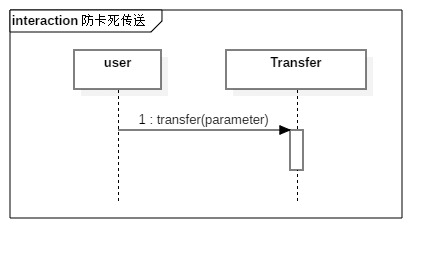
#### 与房间物体互动



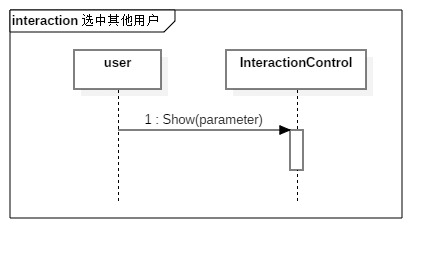
#### 做出动作



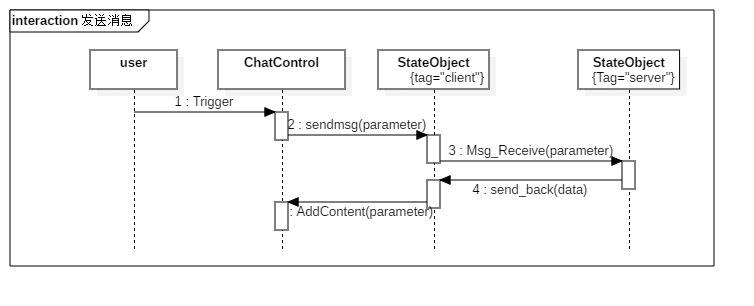
#### 防卡死传送

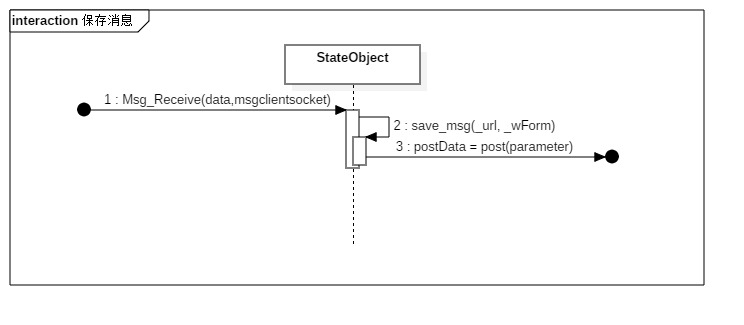


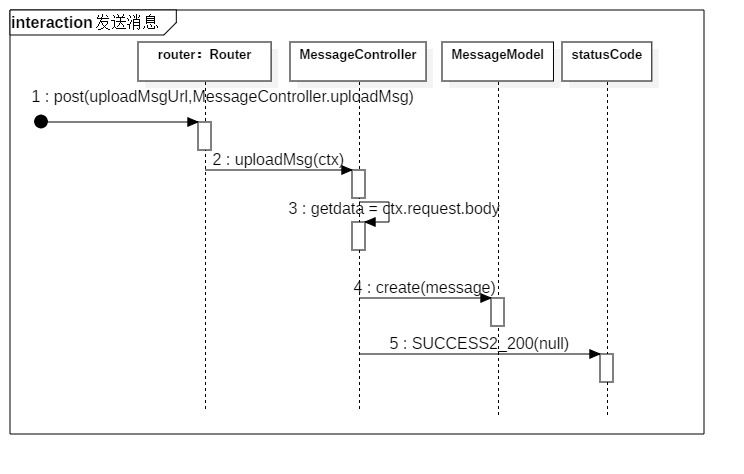
#### 选中其他用户



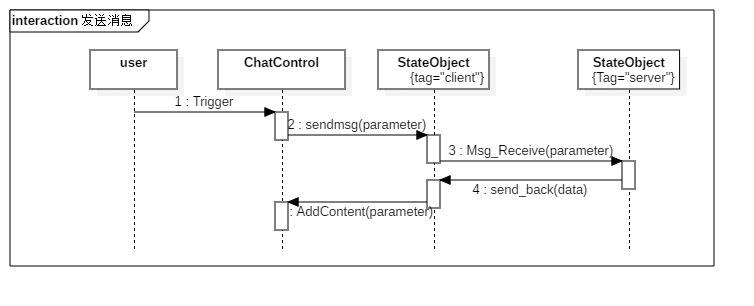
#### 发送消息



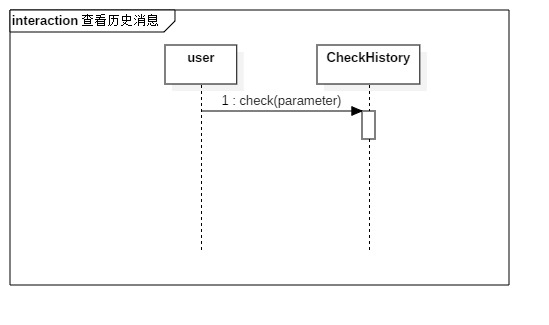




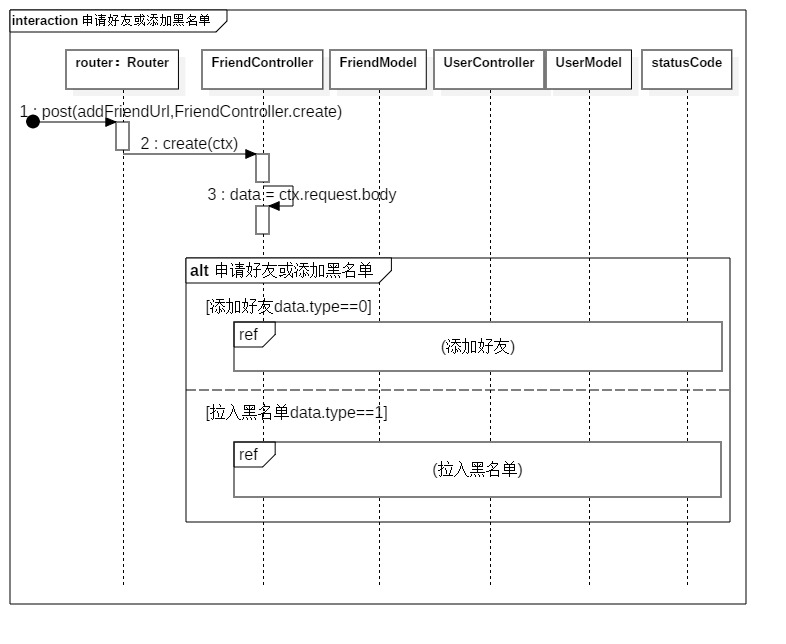
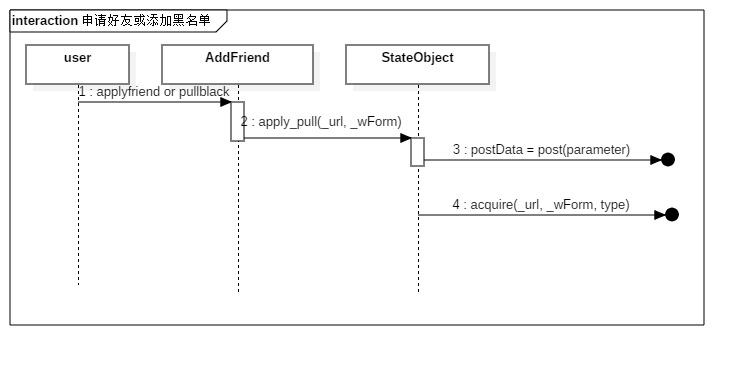
#### 接收消息



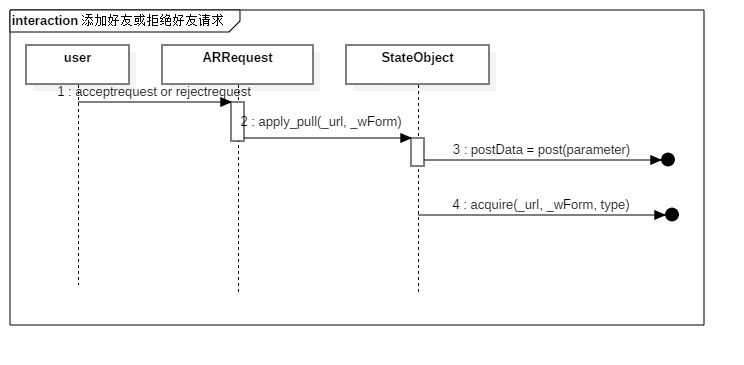
#### 查看历史消息



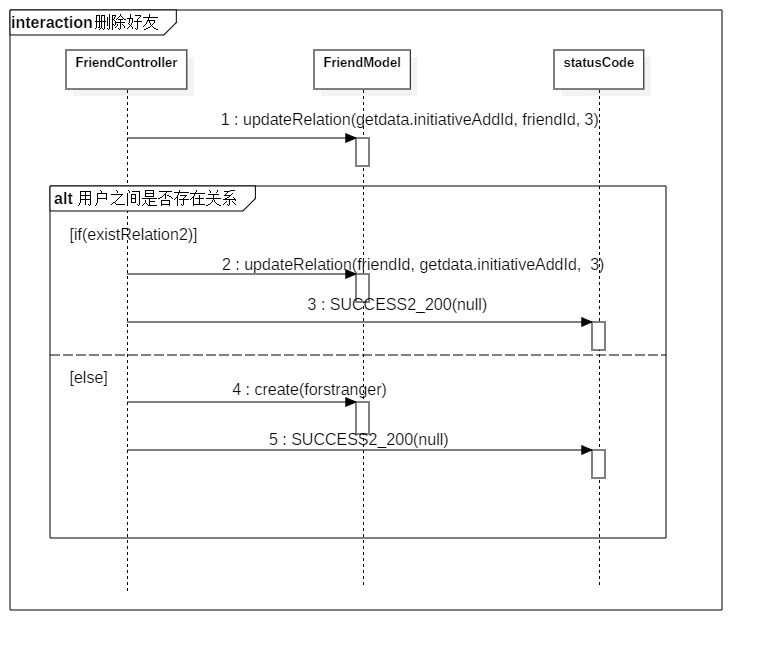
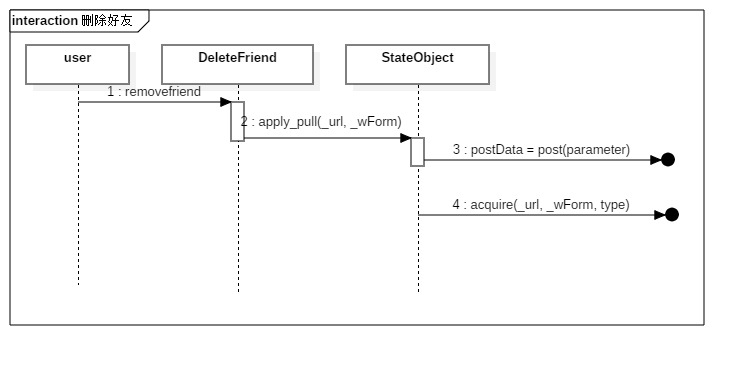
#### 申请好友



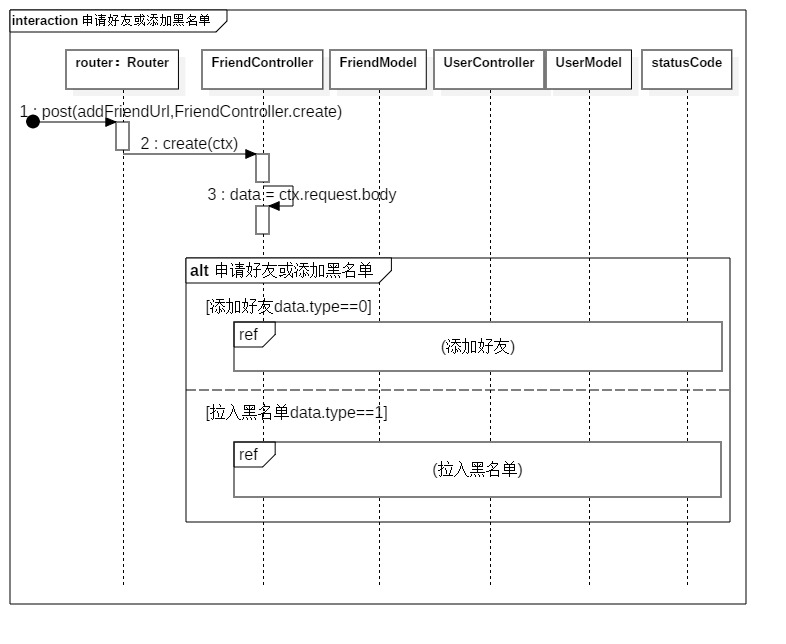
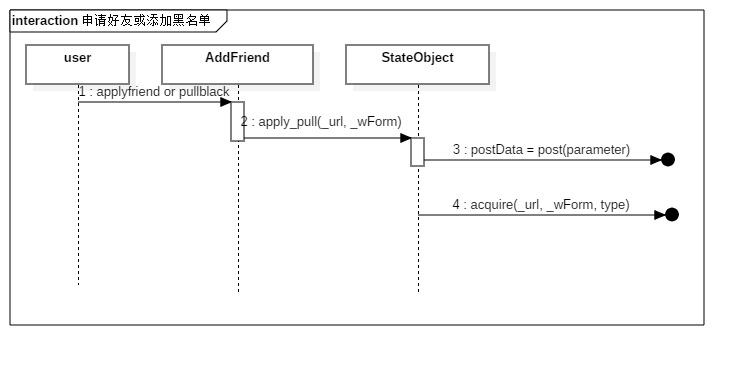
#### 添加好友



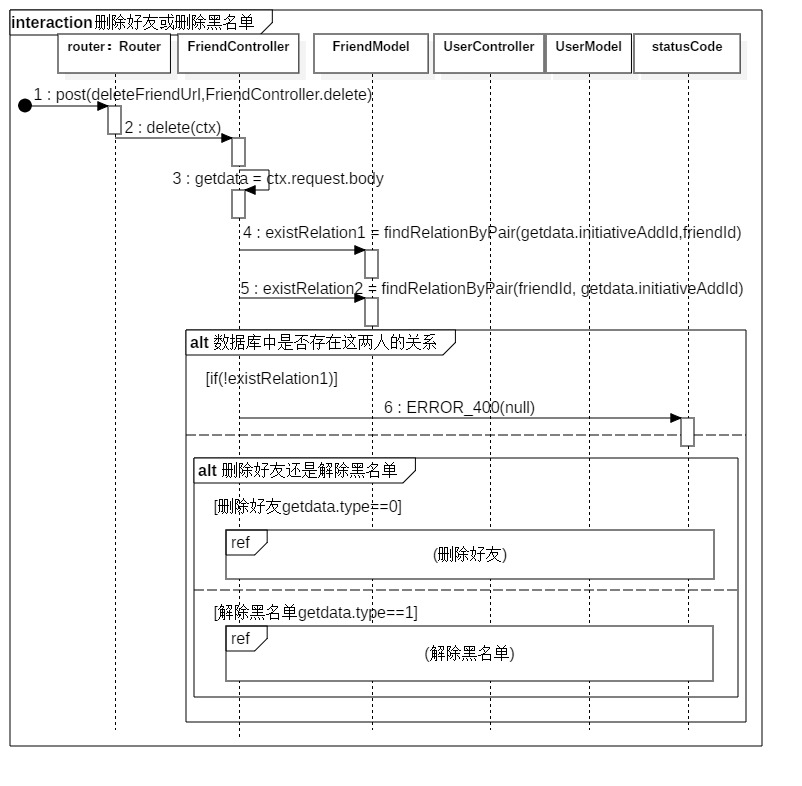
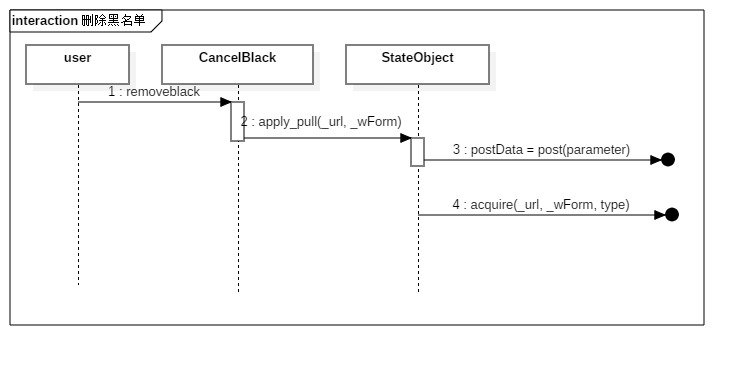
#### 删除好友



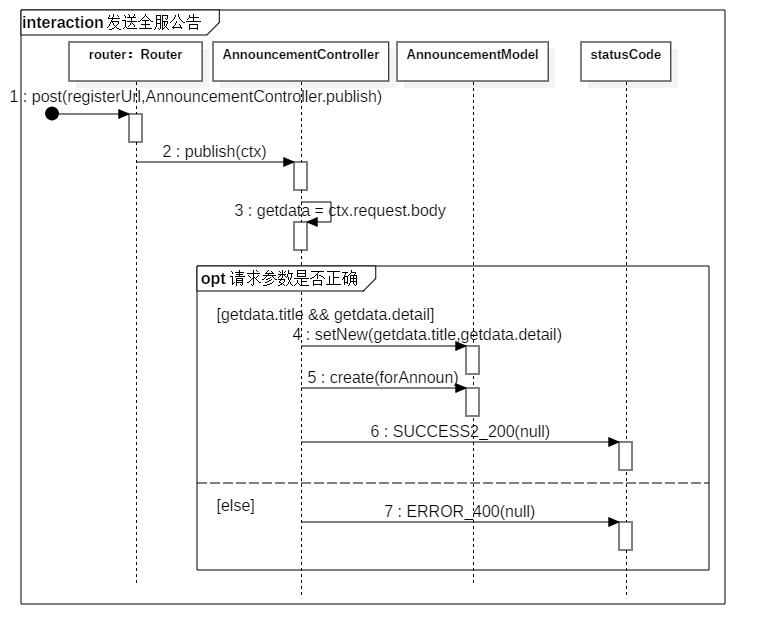
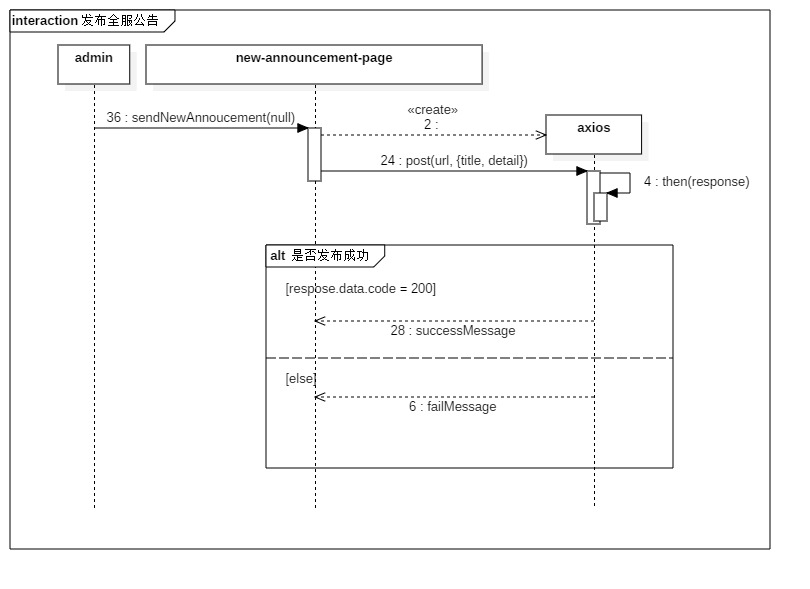
#### 添加黑名单



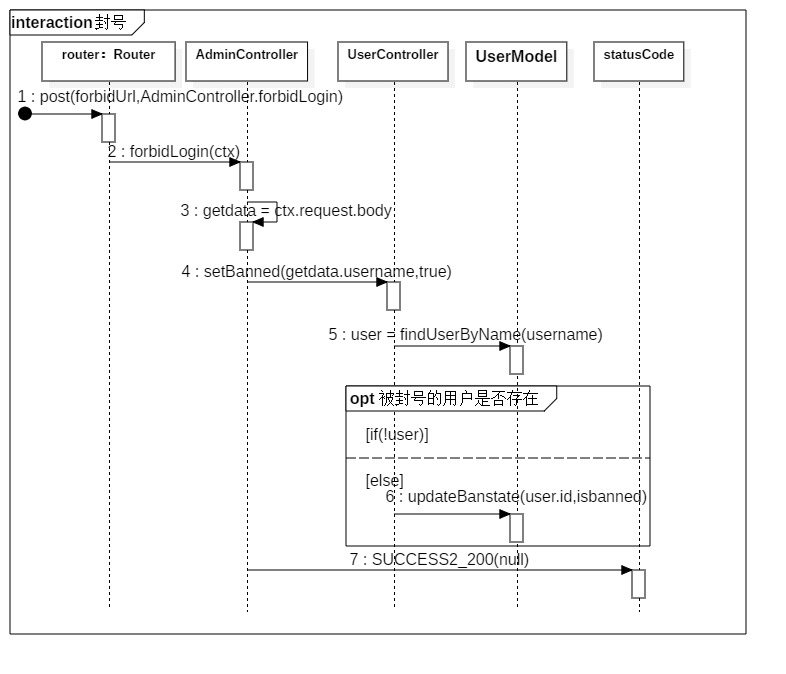
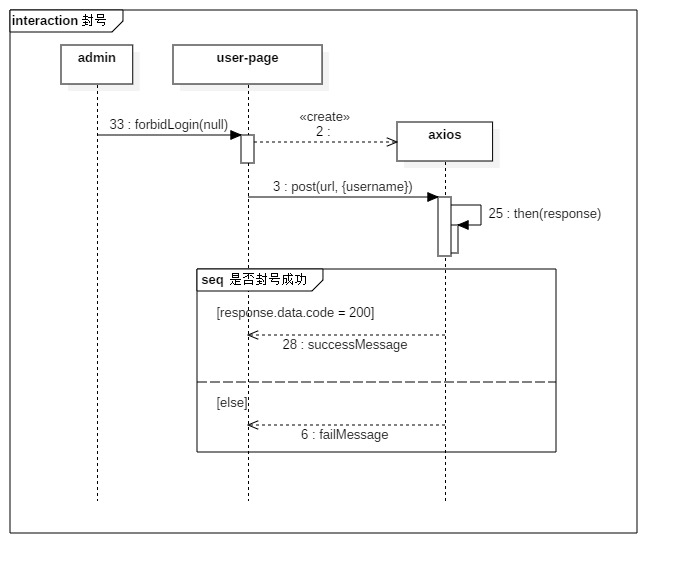
#### 删除黑名单



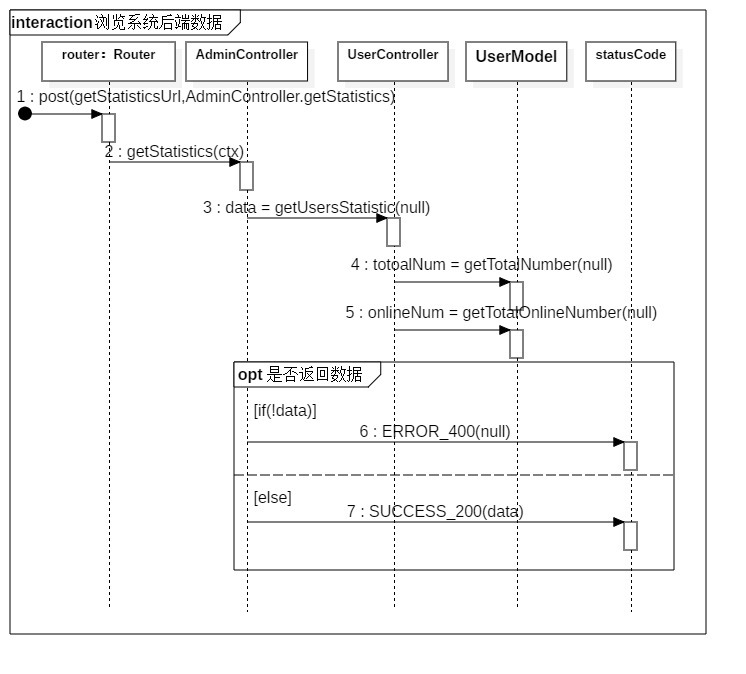
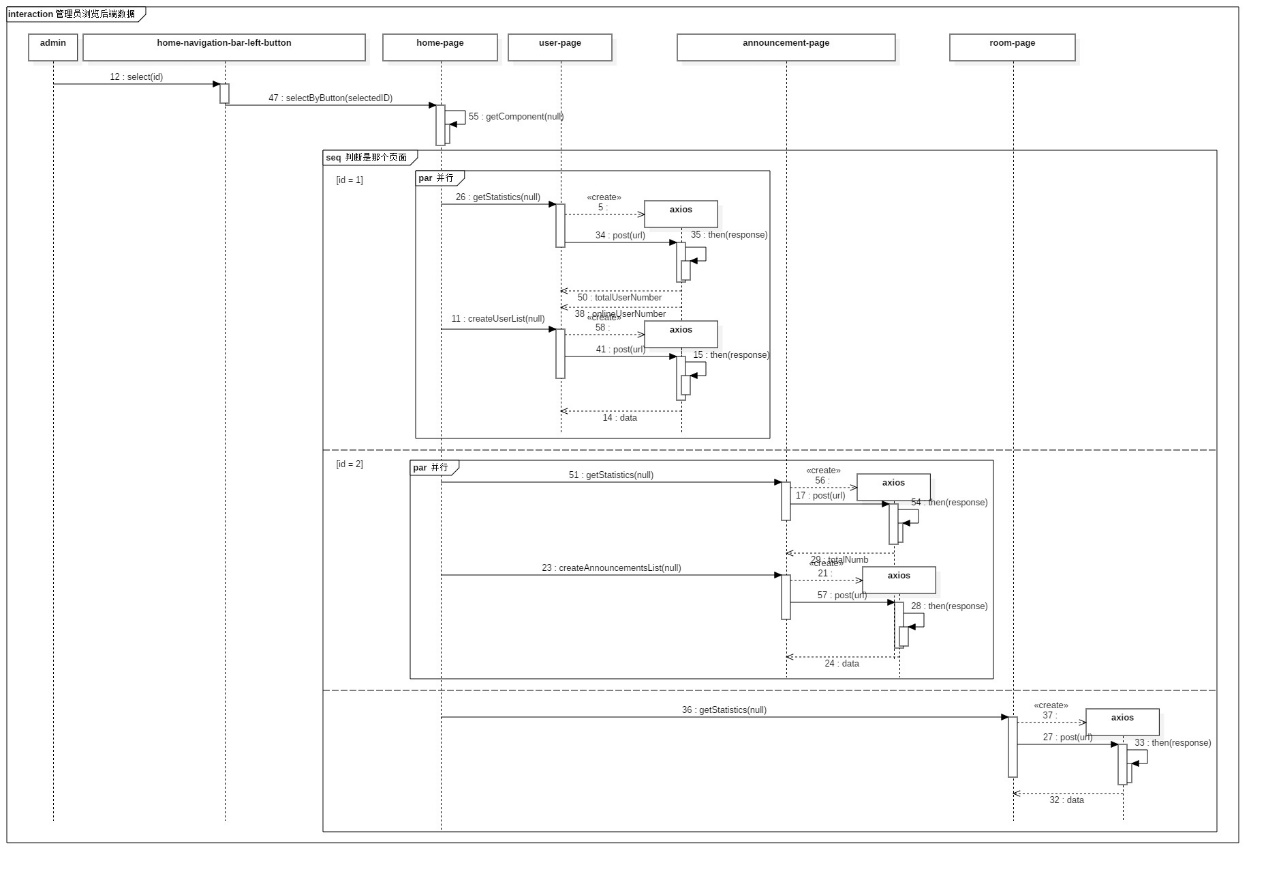
#### 发布全服公告



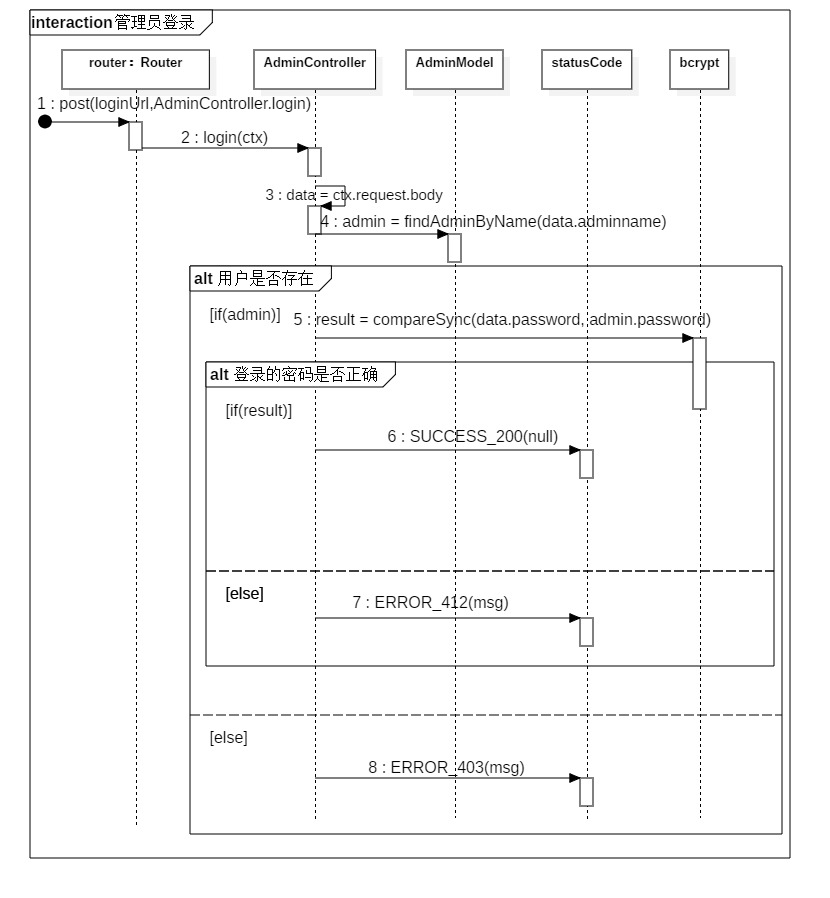
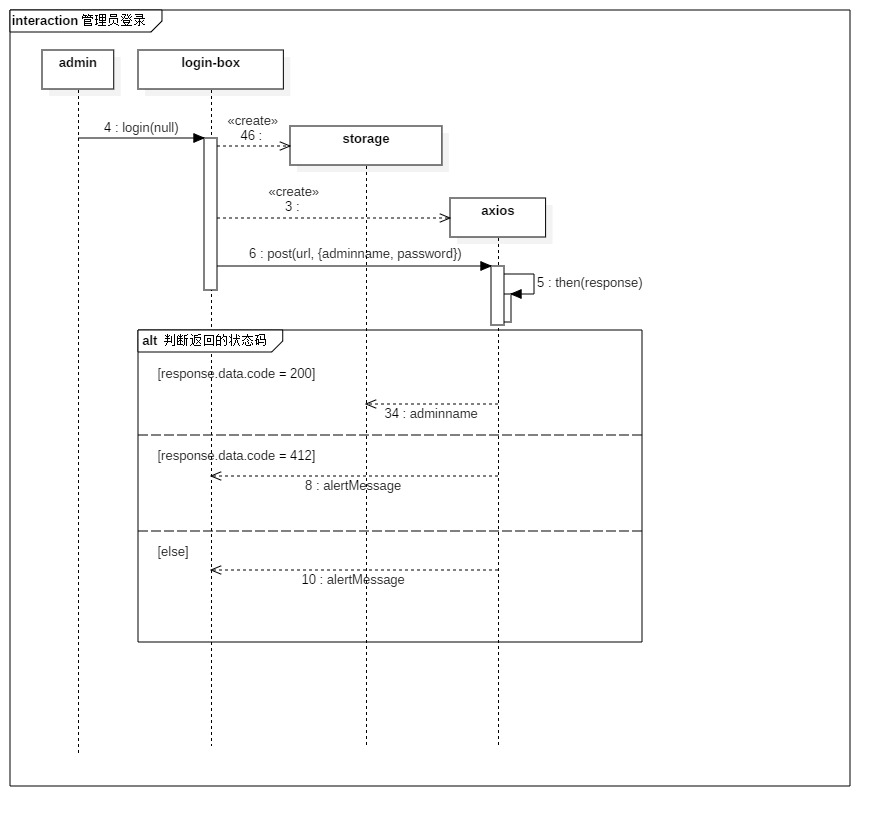
#### 封号



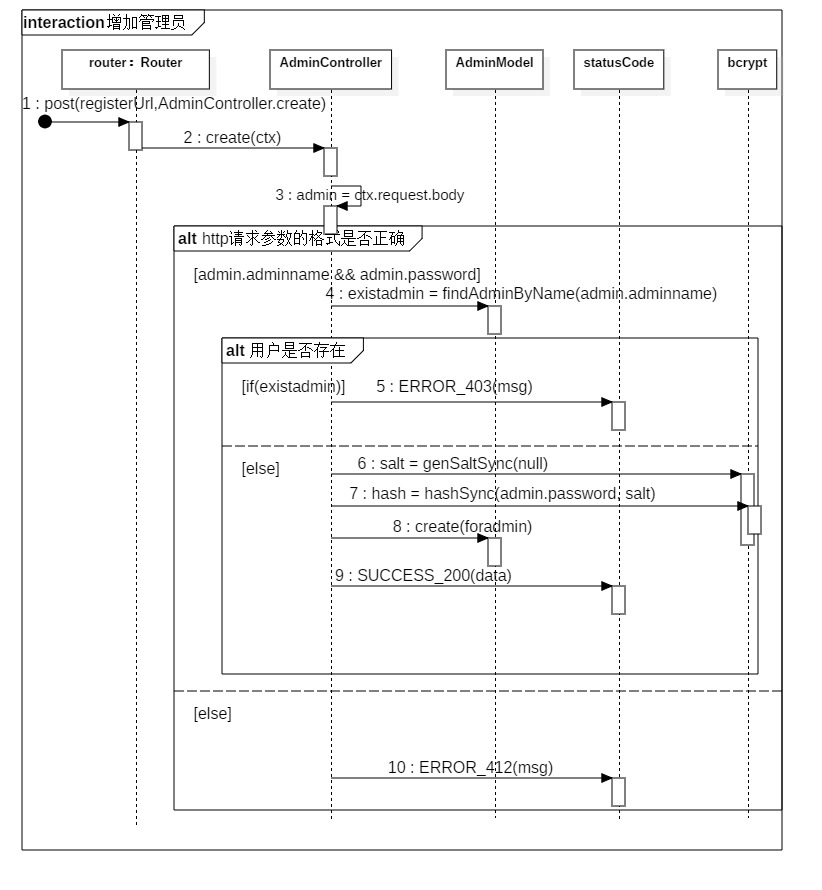
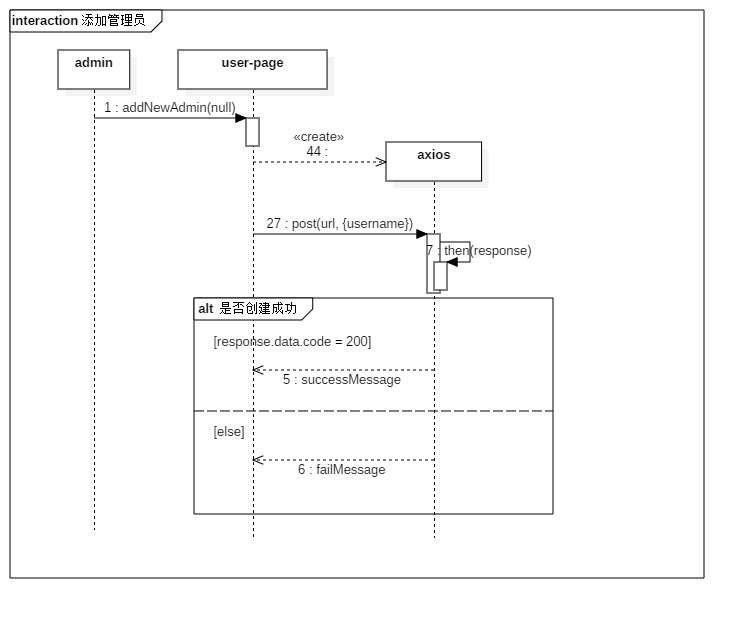
#### 浏览系统后端数据



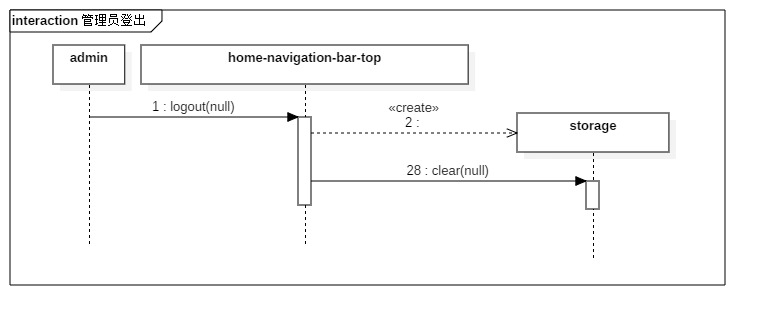
#### 管理员登录



#### 添加管理员



#### 管理员登出



* 1. 系统非功能设计

a) 防止服务端在更新所有用户实时位置时的延迟带来的人物动作卡顿，采用插值的方法来填补画面损失

b) 管理员发布公告不需要很高的即时性，chatstyle服务端可以采用轮询的方式来获取公告，使chatstyle服务端更加专注于其他业务。