**青年租房管理系统**

**[需求规格说明书]**

|  |  |
| --- | --- |
| 成员 | 朱康乐 |
| 成员 | 刘馨文 |
| 成员 | 许澈 |
| 成员 | 李康宁 |
| 成员 | 蒋若曦 |

目录

[1.引言 4](#_Toc102322591)

[1.1 项目背景 4](#_Toc102322592)

[1.2 编写目的 4](#_Toc102322593)

[1.3 系统介绍 4](#_Toc102322594)

[2.现有系统描述 6](#_Toc102322595)

[2.1 角色定义 6](#_Toc102322596)

[2.2 作业流程 6](#_Toc102322597)

[2.3 单据、账本和报表 7](#_Toc102322598)

[2.4 可能的变化 7](#_Toc102322599)

[3.非技术要求 7](#_Toc102322600)

[4.系统环境 8](#_Toc102322601)

[4.1 硬件运行环境 8](#_Toc102322602)

[4.2 软件运行环境 8](#_Toc102322603)

[4.3 开发环境 8](#_Toc102322604)

[5.目标系统功能需求 9](#_Toc102322605)

[6.非功能性需求 39](#_Toc102322606)

[6.1 正确性需求 39](#_Toc102322607)

[6.2 安全性需求 39](#_Toc102322608)

[6.3 界面需求 40](#_Toc102322609)

[6.4 数据要求 40](#_Toc102322610)

[6.5 稳定性需求 41](#_Toc102322611)

[6.6 灵活性需求 41](#_Toc102322612)

[6.7 扩展性需求 41](#_Toc102322613)

[6.8 数据管理能力需求 41](#_Toc102322614)

[6.9 故障处理能力需求 41](#_Toc102322615)

[7．目标系统界面 42](#_Toc102322616)

[7.1界面需求 42](#_Toc102322617)

[7.2界面设计 42](#_Toc102322618)

## 1.引言

### 1.1 项目背景

20世纪90年代住房制度改革以来，中国房地产业发展迅速，不仅显著改善居民居住条件，还在拉动经济增长、促进就业等方面发挥重要作用。

随着房价的上涨，租房市场不断扩大，而青年更是租房的主力。为此，我们考虑设计青年租房管理系统，利用自己的专业知识方便广大青年进行房屋租赁，解决租房信息不对称等问题。

### 1.2 编写目的

本需求规模说明书旨在作为用户和软件开发人员达成的技术协议书，作为着手进行设计工作的基础和依据，同时在系统开发完成以后，为产品的验收提供依据。

### 1.3 系统介绍

#### 1.3.1 系统名称

青年租房管理系统

#### 1.3.2 系统功能

（1）租客注册：绑定手机（或邮箱）并设置个人资料；

（2）顾客租房：租房包括短租与长租，房间类型有单人间、双人间与四人间；

（3）租房流程为：顾客选择长租或短租（短租按天租，长租按月租），查询可租房间并选择，提交租房申请后，客服人员对顾客资料进行审核并决定是否通过，通过后，短租需顾客立刻提交租金，长租则系统自动生成合同，顾客导出打印后双方线下签订，顾客每月最后一星期提交租金，系统自动邮件提醒；

（4）顾客可查询与管理订单，并可进行长租续约；

（5）租客报修与投诉：租客通过上传图片＋文字描述的形式进行报修与投诉；

（6）客服管理租客：客服可查询租客并修改租客资

（7）客服管理合同：客服可查询与管理合同；

（8）客服房间管理：除了线上预约租房外，租客还可当面与前台租房，客服查询可用房间并帮助租房，若顾客未注册需先在平台注册，客服还可添置新房源与暂停出租问题房源；

（9）客服管理工单：客服对报修建立工单并安排师傅，对投诉进行回复；

（10）师傅处理工单：师傅由客服人员线下审核并导入，师傅接到工单后进行线下处理，处理完成后回复工单，租客可对师傅进行评价。

#### 1.3.3 系统使用范围

本系统面向广大有租房需求的青年，包括但不限于毕业大学生。

#### 1.3.4 其他要求

“青年租房管理系统”所面向的用户群体主要为PC端用户（后续将开放针对Android与IOS平台的服务），为保障便捷性，平台为Web服务。由于客户所使用设备分辨率各有不同，平台应保障在不同分辨率下均能提供友好的交互服务。

平台使用范围较广，应保证即便出现可能发生的最大同时访问量，系统仍然能够正常运转，响应时间应在可接受范围内。除此之外，系统还应在安全性、可靠性上予以保障，不应出现数据丢失、隐私泄露等事故。

#### 1.3.5 开发团队说明

|  |  |  |
| --- | --- | --- |
| 姓名 | 职位 | 描述 |
| 朱康乐 | 后端开发，项目总体规划 | 检查各部门项目进度，撰写相关报告文档，后端代码设计与编写，后端请求处理代码编写 |
| 李康宁 | 界面设计，前端开发，前后端交互实现 | 设计网站界面，前端代码设计与编写，前端请求代码编写 |
| 许澈 | 后端开发，数据库设计 | 设计数据库的具体结构，后端代码设计与编写 |
| 蒋若曦 | 界面设计，前端开发，前后端交互实现 | 设计网站界面，前端代码设计与编写，前端请求代码编写 |
| 刘馨文 | 界面设计，前端开发，代码测试 | 设计网站界面，前端代码设计与编写，进行代码测试 |

##### 表1 开发团队说明

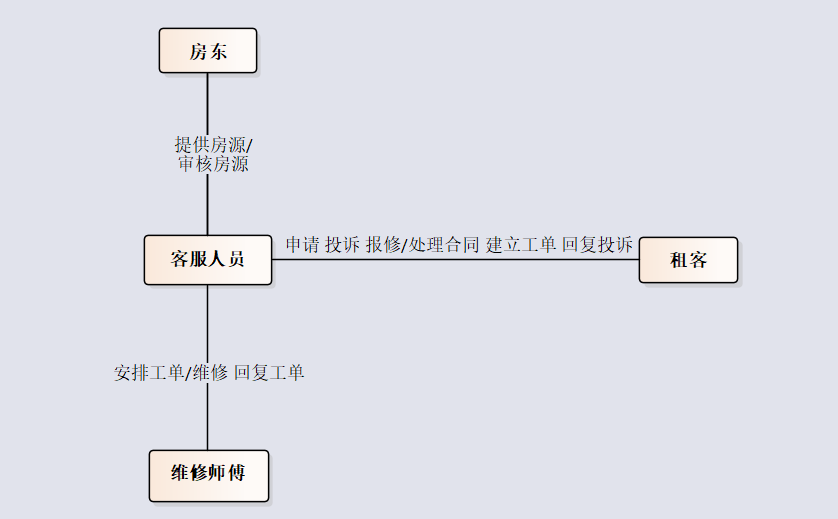
## 2.现有系统描述

### 2.1 角色定义

|  |  |
| --- | --- |
| 编号 | 角色 |
| 00 | 游客 |
| 01 | 用户 |
| 01a | 租客 |
| 01b | 房东 |
| 01c | 客服人员 |
| 01d | 维修师傅 |
| 02 | 系统管理员 |

##### 表2 角色定义

### 2.2 作业流程



##### 图1 作业流程图

### 2.3 单据、账本和报表

暂无

### 2.4 可能的变化

（1）客户的需求可能发生变更。

（2）需求分析中可能存在谬误，有潜在的修改可能。

## 3.非技术要求

（1）本系统第一版工期约为1个月。开发流程为：需求分析、设计、编码实现、单元测试、集成与负载测试、交付及后续运维，其中需求分析的更新贯穿于整个开发过程，以适应用户需求的更改。

（2）要交付的工作产品有：软件开发计划书、需求规格说明书、软件设计说明书、软件测试报告、用户使用手册、项目源代码、可执行程序。

（3）依照软件开发合同，本系统的客户需求可能发生变更。变更时应根据合同约定修改需求，并更新相应文档、设计及实现。

## 4.系统环境

### 4.1 硬件运行环境

（1）服务器

处理器型号：n4系列 III

内存容量：1核2GB

网络配置：1 Mbps

（2）Web浏览PC机

处理器型号：AMD 或Intel 1.6 GHz以上

内存要求：256MB以上

网络配置：100M带宽网络适配器

### 4.2 软件运行环境

（1）服务器

操作系统：Ubuntu 16.04或以上版本系统。

数据库：MySQL 5.0。

（2）客户机

操作系统：无限制

浏览器：支持所有主流浏览器。推荐使用Firefox 或者 Chrome等使用webkit内核的浏览器浏览本网站。

网络环境：客户机应能够访问互联网

### 4.3 开发环境

本软件开发团队使用个人电脑作为开发机，软硬件配置情况如下：

（1）硬件环境

处理器型号：Intel® Core™ i7-7500U CPU @2.70Ghz 2.90Ghz

内存空间：8GB

硬盘容量：500GB

输入输出设备：鼠标、键盘、显示屏

网络配置：100M带宽网络适配器。

（2）软件环境

操作系统：Windows 10 64-bits

数据库：DBeaver 连接 MySQL

开发语言：Python、CSS、HTML、JavaScript等

开发框架：Django 3.2、Vue 3

软件运行环境：.NET 4.5, VC2012

集成开发环境：Pycharm 2021.4、Visual Studio Code 2021

WEB 服务器：Microsoft IIS 6.0或以上版本。

测试工具：Visual Studio

版本控制工具：Git

（注：本小组可能依据实施过程中的实际需要对于软件和硬件环境做出相应的调整）

## 5.目标系统功能需求

“青年租房系统”的功能概述如图2所示。

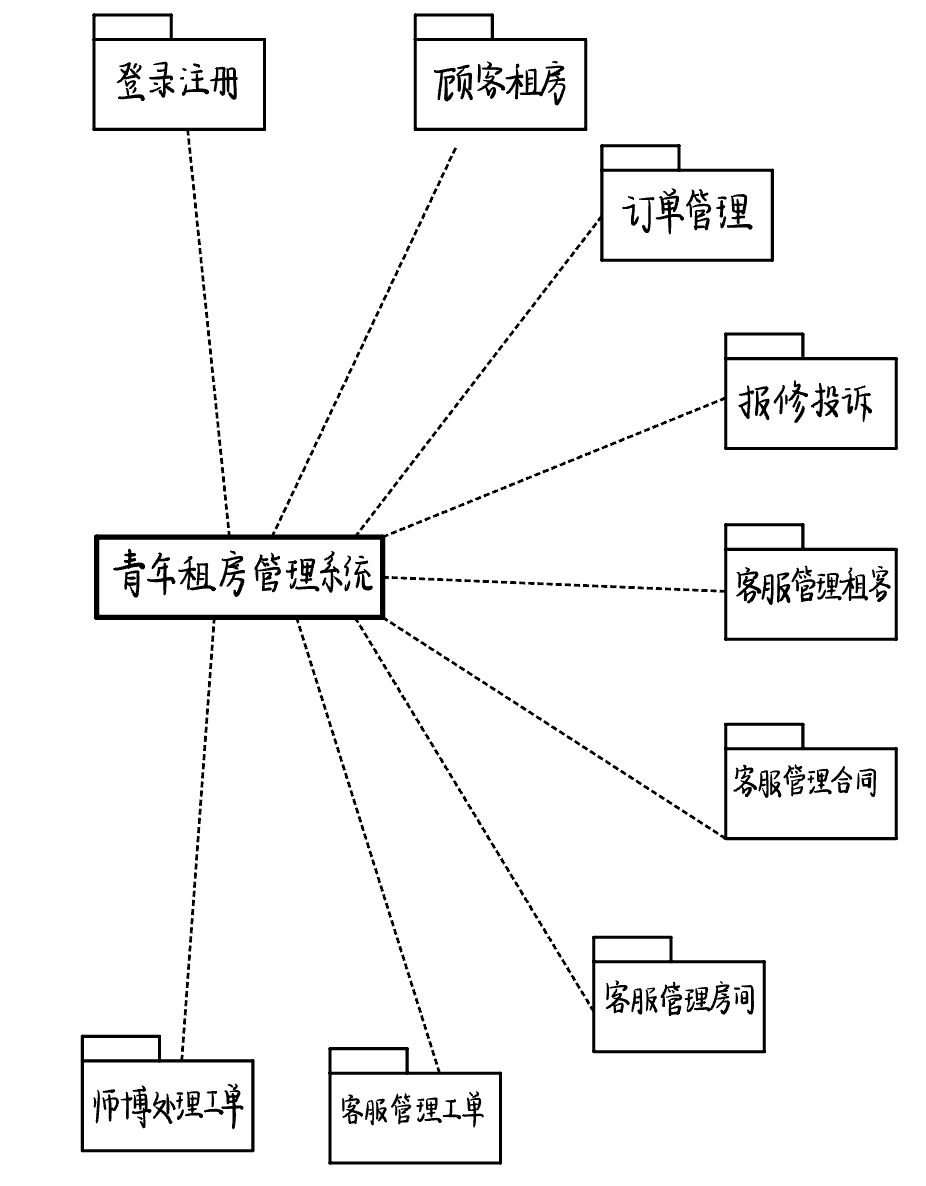
****

图2 “青年租房管理系统”的功能概述

（1）登录注册

用户登录和游客注册的用例图如图3所示。

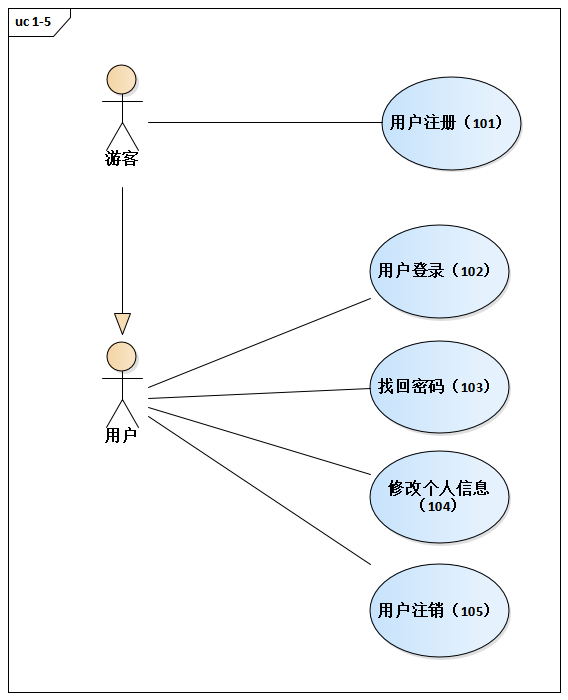


图3 注册登录模块的用例图

**表3 “用户注册”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 101 | **用例名称** | 用户注册 |
| 使用人员 | 游客 | 扩展点 | 无 |
| 输入 | 用户基本信息（手机号或邮箱、其他个人资料） | | |
| 系统响应 | 系统将用户基本信息存入数据库 | | |
| 输出 | 用户可用注册时的用户名和密码登录 | | |
| 前置条件 | 游客申请注册 | 后置条件 | 游客成功注册为用户 |
| 活动步骤 | 1. 游客选择注册。 2. 系统返回注册页面。 3. 游客输入相应信息。 4. 系统验证注册信息。 5. 游客提交注册信息。 6. 系统给游客所填邮箱发送邮件或给所填手机号发送验证码。 7. 用户点击邮箱内链接或填写验证码。 8. 系统提醒注册成功。 | | |
| 异常处理 | 1. 游客输入邮箱或手机号已经被注册，或注册信息与系统验证不一致，系统返回注册页面并给出提示信息。 2. 系统异常，无法注册，给出相应信息，如网站维护。 | | |

**表4 “用户登录”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 102 | **用例名称** | 用户登录 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 用户注册时的用户名和密码 | | |
| 系统响应 | 用户的登陆时间等相关信息存入数据库中 | | |
| 输出 | 相关用户的页面 | | |
| 前置条件 | 该用户必须是本网站已经注册的成员 | 后置条件 | 该用户登录成功 |
| 活动步骤 | 1. 用户选择登录。 2. 系统返回一个登录页面。 3. 用户输入手机号或邮箱、密码并提交。 4. 系统进行系统验证，验证成功，记录该用户为登录用户并返回主页面。 | | |
| 异常处理 | 1. 用户忘记密码，选择“找回密码”功能，进入“找回密码”用例。 2. 系统验证用户登录信息有错，提示用户重新登录。 3. 系统处理异常，系统给出相应的提示信息。 | | |

**表5 “找回密码”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 103 | **用例名称** | 找回密码 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 用户注册时的邮箱号或手机号 | | |
| 系统响应 | 系统根据注册邮箱号或手机号找到相应用户并返回其对应的密码设置页面 | | |
| 输出 | 用户重新设置自己的密码 | | |
| 前置条件 | 该用户必须是本网站已经注册的成员 | 后置条件 | 该用户登录成功 |
| 活动步骤 | 1. 用户选择“找回密码”。 2. 系统返回一个密码找回页面。（要求用户输入注册时的邮箱号或手机号，系统自动发送邮件或验证码到用户的邮箱或者手机，用户输入收到的验证码重新设置密码。） 3. 用户输入新的密码并提交。 4. 系统进行验证，验证成功，提示修改成功并自动跳转至登录页面。 | | |
| 异常处理 | 1. 若用户输入错误的邮箱号或手机号，则系统提示错误并返回登录页面。 2. 重新设置密码时，用户输入验证码不正确，则系统提示验证错误并返回找回密码页面。 3. 系统处理异常，系统给出相应的提示信息。 | | |

**表6 “修改个人信息”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 104 | **用例名称** | 修改个人信息 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 用户输入个人的相关信息 | | |
| 系统响应 | 系统在数据库中用用户现在的个人信息替换以前的个人信息 | | |
| 输出 | 用户的个人信息被修改 | | |
| 前置条件 | 该用户必须是本网站已经注册的成员且已成功登录的用户 | 后置条件 | 该用户修改个人信息成功 |
| 活动步骤 | 1. 用户选择“修改信息”。 2. 系统返回信息修改页面。 3. 用户修改相关信息并提交。 4. 系统进行系统验证，验证成功，提示修改成功。 | | |
| 异常处理 | 1. 系统验证用户输入有误，提示重新输入并返回“修改信息”页面。 2. 系统处理异常，系统给出相应的提示信息。 | | |

**表7 “用户注销”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 105 | **用例名称** | 用户注销 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统自动修改用户在数据库中的相应状态 | | |
| 输出 | 显示用户未登录 | | |
| 前置条件 | 该用户必须是本网站已经注册的成员且已成功登录的用户 | 后置条件 | 用户成功注销 |
| 活动步骤 | 1. 用户选择“注销”。 2. 系统提示用户成功过注销并返回网站首页。 | | |
| 异常处理 | 系统处理异常，系统给出相应的提示信息。 | | |

（2）顾客租房

顾客租房的用例图如图4所示。

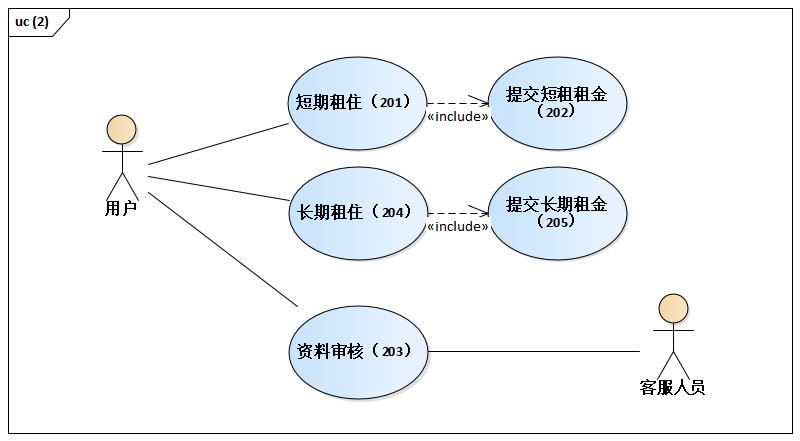


图4 顾客租房模块的用例图

**表8 “短期租住”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 201 | **用例名称** | 短期租住 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统将相应的租房信息添加到缓存和数据库中 | | |
| 输出 | 用户成功完成短租 | | |
| 前置条件 | 该用户必须是本网站已经注册的成员且已成功登录的用户 | 后置条件 | 用户短期租住成功 |
| 活动步骤 | 1. 用户选择“短租”。 2. 返回浏览租住房屋页面。 3. 用户选择房屋，并提交申请。 4. 进入“资料审核”用例。 5. 顾客立刻提交租金。 6. 系统提示用户短租成功并返回网站首页。 | | |
| 异常处理 | 系统处理异常，系统给出相应的提示信息。 | | |

**表9 “提交短租租金”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 202 | **用例名称** | 提交短租租金 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统对数据库中用户租金提交信息进行更新 | | |
| 输出 | 用户成功提交租金 | | |
| 前置条件 | 本网站已经注册的用户且已成功登录的用户提交短租申请审核通过 | 后置条件 | 用户成功提交短租租金 |
| 活动步骤 | 1. 用户进行租金支付。 2. 系统提示租金提交成功 | | |
| 异常处理 | 1. 用户超时未完成支付，系统返回提示信息，并返回提交租金页面 2. 系统处理异常，系统给出相应的提示信息。 | | |

**表10 “资料审核”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 203 | **用例名称** | 资料审核 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统返回审核结果 | | |
| 输出 | 用户资料审核通过 | | |
| 前置条件 | 本网站已经注册的用户且已成功登录的用户提交短期租房申请 | 后置条件 | 用户资料审核通过 |
| 活动步骤 | 1. 客服人员对顾客资料进行审核 2. 资料审核通过，提示审核通过。 | | |
| 异常处理 | 1. 客服人员审核资料不通过，提示用户资料审核不通过，并重新进入“浏览租住房屋”页面。 2. 系统处理异常，系统给出相应的提示信息。 | | |

**表11 “长期租住”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 204 | **用例名称** | 长期租住 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统将相应的租房信息添加到缓存和数据库中 | | |
| 输出 | 用户成功完成长租 | | |
| 前置条件 | 该用户必须是本网站已经注册的成员且已成功登录的用户 | 后置条件 | 用户长期租住成功 |
| 活动步骤 | 1. 用户选择“长租”。 2. 返回浏览租住房屋页面。 3. 用户选择房屋，并提交申请。 4. 进入“资料审核”用例 5. 系统自动生成合同 6. 用户打印合同 7. 双方线下签订。 8. 用户长租成功。 9. 每月交付租金 | | |
| 异常处理 | 1. 客服人员审核资料不通过，提示用户资料审核不通过，并重新进入“浏览租住房屋”页面。 2. 线下签订时双方没有谈妥。 3. 系统处理异常，系统给出相应的提示信息。 | | |

**表12 “提交长租租金”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 205 | **用例名称** | 提交长租租金 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统对数据库中用户租金提交信息进行更新 | | |
| 输出 | 用户成功提交租金 | | |
| 前置条件 | 本网站已经注册的用户且已成功登录的用户已经完成长租 | 后置条件 | 用户成功提交长租租金 |
| 活动步骤 | 1. 系统自动邮件提醒 2. 用户进行租金支付 3. 系统提示租金支付成功 | | |
| 异常处理 | 1. 用户超时未完成支付，系统返回提示信息，并返回提交租金页面 2. 用户没有在每月最后一星期完成房租支付，客服人员会人工提醒。 3. 系统处理异常，系统给出相应的提示信息。 | | |

（3）订单管理

订单管理的用例图如图5所示。

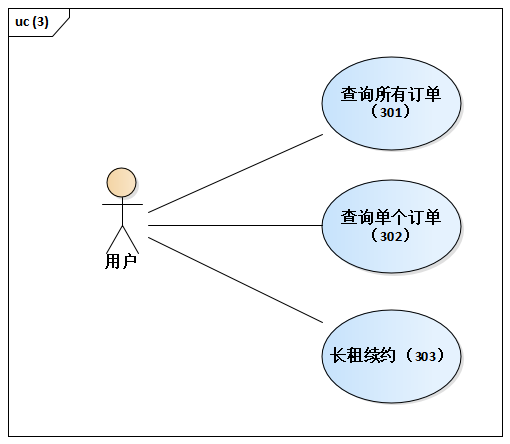


图5 订单管理模块的用例图

**表13 “查询所有订单”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 301 | **用例名称** | 查询所有订单 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统返回订单目录页面 | | |
| 输出 | 系统显示订单目录页面 | | |
| 前置条件 | 1. 该用户为已经成功登录的用户。 2. 该用户在系统中完成过租房。 | 后置条件 | 用户查看所有订单成功 |
| 活动步骤 | 1. 用户点击“查看订单”项。 2. 系统返回订单目录页面。 | | |
| 异常处理 | 1. 系统提示无订单。 2. 系统处理异常，系统给出相应的提示信息。 | | |

**表14 “查询单个订单”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 302 | **用例名称** | 查询单个订单 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统返回相应的订单信息页面 | | |
| 输出 | 系统显示相应的订单信息 | | |
| 前置条件 | 1. 该用户为已经成功登录的用户。 2. 该用户在系统中完成过租房。 | 后置条件 | 用户查看该订单成功 |
| 活动步骤 | 1. 用户点击相应的订单。 2. 系统返回该订单的详细信息。 | | |
| 异常处理 | 系统处理异常，系统给出相应的提示信息。 | | |

**表15 “长租续约”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 303 | **用例名称** | 长租续约 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 系统自动修改数据库中相应信息 | | |
| 输出 | 长租续约成功 | | |
| 前置条件 | 1. 该用户为已经成功登录的用户。 2. 该用户有没有结束的长租订单 | 后置条件 | 用户续约成功 |
| 活动步骤 | 1. 用户选择对应长租订单续约。 2. 出租人确认用户续约请求。 3. 系统提示用户续约成功。 | | |
| 异常处理 | 1. 出租人未同意用户续约请求，系统给出相应提示。 2. 系统处理异常，系统给出相应的提示信息。 | | |

（4）报修投诉

报修投诉的用例图如图6所示。

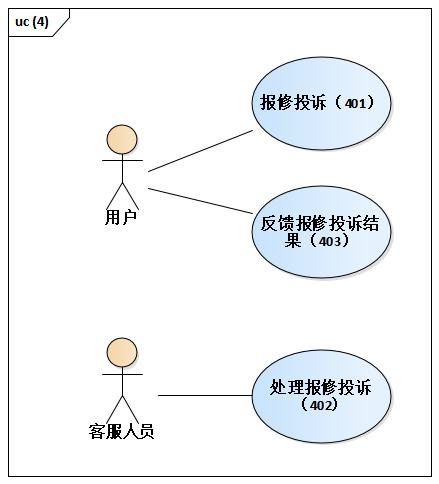


图6 报修投诉模块的用例图

**表16 “报修投诉”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 401 | **用例名称** | 报修投诉 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 图片和文字描述 | | |
| 系统响应 | 用户报修投诉等相关信息存入数据库中 | | |
| 输出 | 成功完成报修投诉 | | |
| 前置条件 | 1. 该用户为已经成功登录的用户。 2. 该用户在系统里完成过租房。 | 后置条件 | 用户报修投诉成功 |
| 活动步骤 | 1. 用户选择报修投诉。 2. 系统返回报修投诉页面。 3. 用户上传图片和输入文字描述。 4. 系统提示用户报修投诉成功 。 | | |
| 异常处理 | 系统处理异常，系统给出相应的提示信息。 | | |

**表17 “处理报修投诉”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 402 | **用例名称** | 处理报修投诉 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 无 | | |
| 系统响应 | 更新数据库中用户报修投诉等相关信息 | | |
| 输出 | 成功处理报修投诉 | | |
| 前置条件 | 用户在系统里提交了报修投诉。 | 后置条件 | 处理报修投诉成功 |
| 活动步骤 | 1. 客服人员查看报修投诉信息。 2. 客服人员联系出租人解决对应问题。 3. 系统将对应报修投诉记录状态标记为“已” | | |
| 异常处理 | 系统处理异常，系统给出相应的提示信息。 | | |

**表18 “反馈报修投诉结果”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 403 | **用例名称** | 反馈报修投诉结果 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 无需输入 | | |
| 系统响应 | 更新数据库中用户报修投诉等相关信息 | | |
| 输出 | 成功完成反馈 | | |
| 前置条件 | 1. 该用户为已经成功登录的用户。 2. 该用户在系统里完成过租房。 3. 该用户在系统里提交过报修投诉。 4. 出租人对相关问题整治完毕。 | 后置条件 | 用户反馈结果成功 |
| 活动步骤 | 1. 用户选择查看报修投诉记录 2. 对相应的报修投诉记录选择“已解决” 3. 系统提示用户反馈成功 | | |
| 异常处理 | 1. 用户报修投诉问题没有得到解决，选择“仍未解决”，重新进入“处理报修投诉”样例。 2. 系统处理异常，系统给出相应的提示信息。 | | |

1. 客服管理租客

客服管理租客的用例图如图7所示。

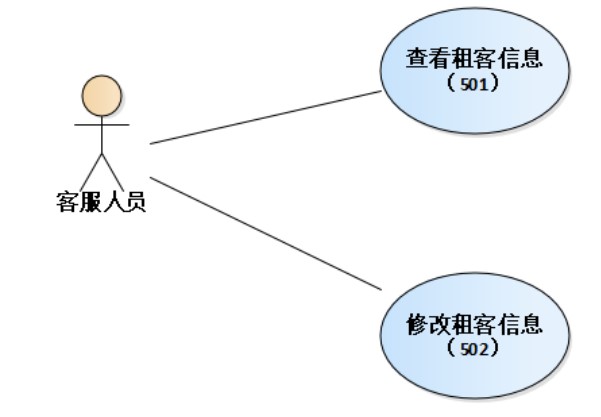


图7 客服管理租客模块的用例图

**表19 “查询用户信息”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 501 | **用例名称** | 查询用户信息 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 租客的用户ID | | |
| 系统响应 | 返回用户的信息页面 | | |
| 前置条件 | 该用户需存在 | 后置条件 | 客服能够查看搜索结果 |
| 交互行为 | 1. 客服通过交互接口输入用户ID，提交 2. 系统校验输入的用户ID合法，查找数据库，查找对应用户，返回对应信息页面 | | |
| 异常处理 | 1 系统验证客服身份，确认其未登录，跳转至登录页面，提示登录  2 系统校验提交信息不合法，返回错误提示  3 系统未找到对应用户，返回错误提示 | | |
| 特别说明 | 无 | | |

**表20 “修改租客信息”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 502 | **用例名称** | 修改用户信息 |
| 使用人员 | 客服人员 | 扩展点 | 可以修改基础信息，还可以修改账号或密码 |
| 输入 | 输入用户的相关信息 | | |
| 系统响应 | 1.系统在数据库中用户现在的个人信息替换以前的个人信息  2. 系统成功修改个人信息，并跳转至个人信息页面 | | |
| 前置条件 | 客服已经登录 | 后置条件 | 成功修改个人信息 |
| 交互行为 | 1. 客服输入用户ID，查找该用户 2. 系统查找数据库，查找对应用户，返回对应信息页面 3. 客服发起修改该用户信息请求 4. 系统返回修改个人信息页面 5. 客服输入相应信息，提交 6. 系统验证提交修改信息，验证成功，予以上传 7. 系统保存上传的数据，返回修改个人信息成功提示 | | |
| 异常处理 | 1. 系统验证客服登录状态，确认客服未登录，跳转至登录页提醒客服登录 2. 系统验证提交修改信息，验证失败，返回错误提示，与对应数据错误一致 | | |
| 特别说明 | 用户昵称，微信号等个人信息数据均为可选数据 | | |

1. 客服管理合同

客服管理合同的用例图如图8所示。

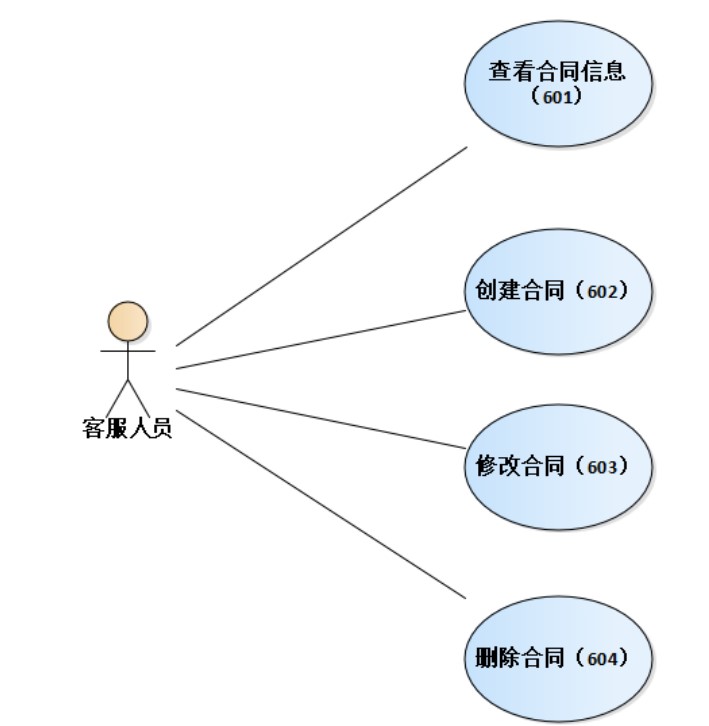


图8 客服管理合同模块的用例图

**表21 “查询合同信息”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 601 | **用例名称** | 查询合同信息 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 合同ID | | |
| 系统响应 | 返回合同详情页面，呈现合同详情数据 | | |
| 前置条件 | 客服已登录 | 后置条件 | 客服能够查看合同详情 |
| 交互行为 | 1客服人员通过交互接口输入合同ID，提交  2系统校验输入的合同ID合法，查找数据库，查找对应合同，返回对应信息页面 | | |
| 异常处理 | 1 系统验证客服人员身份，确认其未登录，跳转至登录页面，提示登录  2 系统校验提交信息不合法，返回错误提示  3 系统未找到对应合同，返回错误提示 | | |
| 特别说明 | 无 | | |

**表22 “创建合同”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 602 | **用例名称** | 创建合同 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 合同ID、用户ID、租金、定金、租赁时间 | | |
| 系统响应 | 在数据库中创建新合同，并提示客服创建成功 | | |
| 前置条件 | 客服人员需处在登录状态，用户已注册 | 后置条件 | 客服成功创建合同，并获知创建成功提示 |
| 交互行为 | 1. 客服人员通过交互页面发起请求 2. 系统校验客服人员身份，返回创建合同页面 3. 客服人员在创建合同页面输入相关信息，提交 4. 系统校验客服人员输入信息，校验成功，上传 5. 系统保存客服人员输入信息，跳转至商品详情页面，返回创建成功提示 | | |
| 异常处理 | 1. 系统验证客服身份，确认其未登录，跳转至登录页，提示客服登录 2. 系统校验客服输入信息不合法，返回错误提示 | | |
| 特别说明 | 无 | | |

**表23 “修改合同”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 603 | **用例名称** | 修改合同 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 合同ID、用户ID、租金、定金、租赁时间 | | |
| 系统响应 | 系统成功修改合同信息，返回修改成功提示消息 | | |
| 前置条件 | 合同需要存在 | 后置条件 | 合同信息得到修改，同时客服能够收到修改成功提示 |
| 交互行为 | 1. 客服通过交互界面发起请求 2. 系统校验客服身份，确认客服已登录，继续 3. 系统查找对应合同，找到合同，返回更改商品页面 4. 客服在更改合同页面输入相关信息，提交 5. 系统校验客服输入信息，合法，上传 6. 系统保存客服更改，跳转回商品详情页面，返回更改成功提示 | | |
| 异常处理 | 1. 系统确认客服未登录，跳转至登录页面提示客服登录 2. 系统未找到对应合同，返回错误提示，提示需求不存在 3. 系统校验客服输入信息，不合法，返回错误提示信息 | | |
| 特别说明 | 修改商品需提交的参数除商品ID外，均为可选 | | |

**表24 “删除合同”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 604 | **用例名称** | 删除合同 |
| 使用人员 | 客服 | 扩展点 | 无 |
| 输入 | 合同ID，删除原因 | | |
| 系统响应 | 返回修改成功提示 | | |
| 前置条件 | 合同需要存在，客服处在登录状态 | 后置条件 | 客服能够收到删除成功提示 |
| 交互行为 | 1. 客服通过交互界面发起请求 2. 系统校验客服登录状态，确认客服已登录，继续 3. 系统查找到合同，继续 4. 系统跳转至合同详情页面，返回删除成功提示 | | |
| 异常处理 | 1. 系统校验客服登录状态，确认客服未登录，跳转至登录页，提示客服登录 2. 系统未找到指定合同，返回错误提示信息，提示未找到 | | |
| 特别说明 | 无 | | |

1. 客服管理房间

客服管理房间的用例图如图9所示。

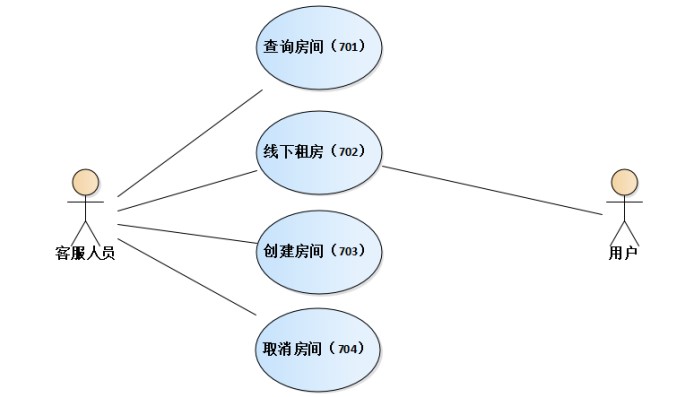


图9 客服管理房间模块的用例图

**表25 “查询房间”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 701 | **用例名称** | 查询房间 |
| 使用人员 | 客服 | 扩展点 | 无 |
| 输入 | 搜索条件 | | |
| 系统响应 | 跳转至可用的房间列表页面，呈现查询结果 | | |
| 前置条件 | 客服已登录 | 后置条件 | 客服能够查看搜索结果 |
| 交互行为 | 1.客服通过交互接口输入搜索条件，提交  2.系统校验输入的条件合法，查找数据库，返回可用房间列表页面，呈现查询结果 | | |
| 异常处理 | 1 系统验证客服身份，确认其未登录，跳转至登录页面，提示登录  2 系统校验提交信息不合法，返回错误提示 | | |
| 特别说明 | 应提供排序与分页机制 | | |

**表26 “线下租房”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 702 | **用例名称** | 查询房间 |
| 使用人员 | 客服（卖方） | 扩展点 | 无 |
| 输入 | 用户名、房间号 | | |
| 系统响应 | 在数据库中创建相关订单，并提示租房成功 | | |
| 前置条件 | 客服已登录、用户已注册 | 后置条件 | 用户成功租房，并返回成功提示 |
| 交互行为 | 1. 客服通过交互接口输入用户名，提交  2. 系统校验用户身份，返回租房页面  3. 客服在租房页面输入相关信息，提交  4. 系统校验输入信息，校验成功，上传 | | |
| 异常处理 | 1 系统验证客服身份，确认其未登录，跳转至登录页面，提示登录  2 系统验证用户身份，确认其未注册，跳转至注册页面，提示注册 | | |
| 特别说明 | 客服在执行该用例过程中的角色为“卖方”  用户在执行该用例过程中的角色为“买方” | | |

**表27 “创建房间”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 703 | **用例名称** | 创建房间 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 房间号、其他房间信息 | | |
| 系统响应 | 在数据库中创建新房间，并提示客服创建成功 | | |
| 前置条件 | 客服需处在登录状态 | 后置条件 | 客服成功创建房间，并获知创建成功提示 |
| 交互行为 | 1.客服通过交互页面发起请求  2.系统校验客服身份，返回创建房间页面  3.客服在创建房间页面输入相关信息，提交  4.系统校验客服输入信息，校验成功，上传  5.系统保存客服输入信息，跳转至房间详情页面，返回创建成功提示 | | |
| 异常处理 | 1.系统验证客服身份，确认其未登录，跳转至登录页，提示客服登录  2.系统校验客服输入信息不合法，返回错误提示 | | |
| 特别说明 | 无 | | |

**表28 “取消房间”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 704 | **用例名称** | 取消房间 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 房间ID，取消出租原因 | | |
| 系统响应 | 返回取消成功提示 | | |
| 前置条件 | 房间需要存在，客服处在登录状态 | 后置条件 | 客服能够收到删除成功提示 |
| 交互行为 | 1.客服通过交互界面发起取消房间请求  2.系统校验客服登录状态，确认客服已登录，继续  3.系统查找到指定房间，返回取消房间页面，继续  4.客服在取消房间页面输入取消原因，提交  4.系统保存客服输入信息，将房间状态更新为暂停出租，跳转至房间详情页面，返回取消成功提示 | | |
| 异常处理 | 1.系统校验客服登录状态，确认客服未登录，跳转至登录页，提示客服登录  2.系统未找到指定合同，返回错误提示信息，提示未找到 | | |
| 特别说明 | 注意暂停出租房间不是删除房间，相关数据仍然应该存在在系统中 | | |

1. 客服管理工单

客服管理工单的用例图如图10所示。

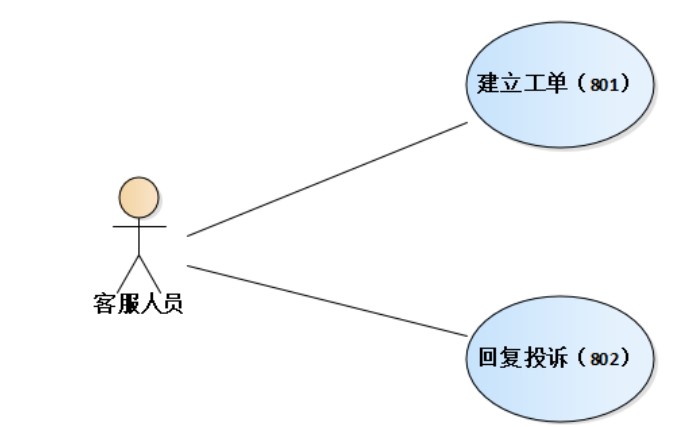


图10 客服管理工单模块的用例图

**表29 “建立工单”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 801 | **用例名称** | 建立工单 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 报修时间、报修原因、报修地点、报修用户ID、师傅ID | | |
| 系统响应 | 在数据库中创建新工单，并提示客服创建成功 | | |
| 前置条件 | 客服需处在登录状态 | 后置条件 | 客服成功创建工单，并获知创建成功提示。师傅收到工单详细信息。 |
| 交互行为 | 1.客服发起建立工单请求  2.系统校验客服身份，返回创建工单页面  3.客服在创建工单页面输入报修的相关信息，提交  4.系统校验客服输入信息，校验成功，上传  5.系统查找师傅ID，并验证是有效的发送对象  6.系统将工单发送给师傅  7.系统保存客服输入信息，跳转至工单详情页面，返回创建成功提示 | | |
| 异常处理 | 1.系统校验客服是否登录，跳转至登录页，提示客服登录  2.系统校验客服输入信息不合法，返回错误提示  3.若目标师傅不存在或者无法发送消息，返回错误提示 | | |
| 特别说明 | 无 | | |

**表30 “回复投诉”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 802 | **用例名称** | 回复投诉 |
| 使用人员 | 客服人员 | 扩展点 | 无 |
| 输入 | 无 | | |
| 系统响应 | 系统向相关用户发送回复 | | |
| 前置条件 | 客服处在登录状态 | 后置条件 | 相关用户收到回复 |
| 交互行为 | 1. 客服发起回复投诉请求 2. 系统校验客服身份，确认其已登录，查找一组投诉数据，返回回复投诉页面，呈现投诉信息 3. 客服在回复投诉页面输入回复，提交 4. 系统查找用户ID，并验证是有效的发送对象 5. 系统将回复发送给目标用户 6. 系统保存客服输入信息，跳转至投诉详情页面，返回成功提示 | | |
| 异常处理 | 1.系统校验客服登录状态，确认客服未登录，跳转至登录页，提示客服登录  2.若投诉用户不存在或者无法发送消息，返回错误提示 | | |
| 特别说明 | 搜索投诉需要支持分页与排序 | | |

1. 师傅处理工单

师傅处理工单的用例图如图11所示。

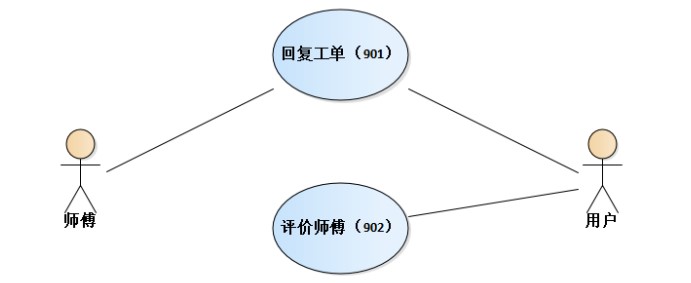


图11 师傅处理工单模块的用例图

**表31 “回复工单”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 901 | **用例名称** | 回复工单 |
| 使用人员 | 师傅，用户 | 扩展点 | 无 |
| 输入 | 回复工单的内容 | | |
| 系统响应 | 系统向相关用户发送回复 | | |
| 前置条件 | 师傅处在登录状态 | 后置条件 | 相关用户收到回复，工单修改成功 |
| 交互行为 | 1.师傅发起回复工单请求  2.系统校验师傅身份，确认其已登录，查找一组工单数据，返回回复工单页面，呈现工单信息  3.客服在回复工单页面输入回复，提交  4.系统查找用户ID，并验证是有效的发送对象  5.系统将回复发送给目标用户  6.系统保存师傅输入信息，跳转至工单详情页面，返回成功提示 | | |
| 异常处理 | 1.系统校验师傅登录状态，确认师傅未登录，跳转至登录页，提示师傅登录  2.若投诉用户不存在或者无法发送消息，返回错误提示 | | |
| 特别说明 | 无 | | |

**表32 “评价师傅”用例**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | 902 | **用例名称** | 评价师傅 |
| 使用人员 | 用户 | 扩展点 | 无 |
| 输入 | 对师傅的评价 | | |
| 系统响应 | 将评价写入数据库，返回评价成功提示 | | |
| 前置条件 | 注册用户已登录，且为工单中的反馈方 | 后置条件 | 评价被写入数据库中并与工单关联，同时用户能够看到评分成功提示 |
| 交互行为 | 1. 用户通过交互式页面输入评价，提交 2. 系统根据师傅ID更新师傅的评分，返回评分成功提示 | | |
| 异常处理 | 1. 系统未找到对应工单，返回错误提示 | | |
| 特别说明 | 师傅ID已封装在工单中，无须用户手动输入 | | |

## 6.非功能性需求

### 6.1 正确性需求

系统正确性需求主要包括如下几项：

1. 能够为用户提供正确的注册、注销、登录机制，用户在注册之后能正常使用系统。
2. 用户能够准确无误地选择目标房型，发起、查询并管理订单，进行申诉等。
3. 客服能够准确无误即时收到用户的需求，并对用户的需求以及系统其它问题准确管理。
4. 师傅能够准确收到并处理客户反馈的问题。
5. 房东能够准确上传房间信息。
6. 系统用户在系统中进行的任何数据修改操作均应达成其对应目的．

### 6.2 安全性需求

用户在系统上注册的信息涉及个人隐私，所以存储这些信息的数据库应具有很高的安全性。为了尽量提高数据的安全性，我们使用框架的官方API来进行前后端的数据传输和对数据库的访问。本地数据存储采用键值对的方式存放在数据库中，关键信息完全由后台进行配置。本系统应该能够记录系统运行时发生的所有错误和引发该错误的操作。

本系统严格采用编码规范进行开发，使用B/S体系架构。通过一系列的测试手段来完善系统，保证系统的可靠性。

### 6.3 界面需求

由于本系统基于Django与Vue框架，网站界面应简洁、清晰，但不失美观；颜色搭配应合理，色调柔和，各页面主题风格一致。

系统需要通过界面与用户进行交互，相应的交互接口应当方便易用。

此外，用户的显示分辨率可能有所不同，系统的界面应在不同分辨率下均能够正常显示，并保证美观性和易用性。

### 6.4 数据要求

对数据无特殊要求，但要注意部分数据的越界问题，如用户发布的商品数、商品价格等。

一些通性要求如下所示：

1. 由于面向人民币区，基于基本货币制度，租房价格保留小数点后两位，商品价格必须为非负数。
2. 时间/日期呈现统一采用XXXX-XX-XX(XX:XX)格式显示，年份使用四位，月份两位，日期两位。小时两位，使用二十四小时制，分钟两位。以上所有数值在不足位数时使用零补位。此外，除非有必须需要，否则秒数不予显示。
3. 邮箱满足规范格式。手机号满足十一位标准格式。
4. 密码长度6-16位，仅应包含数字、英语字母、以及可见标点符号。密码区分大小写，不支持Unicode字符，不支持全角字符。

### 6.5 稳定性需求

该系统部署后，在硬件条件和支持软件条件没有发生变化的情况下，能够一直保持正常运行状态，直到系统被升级或替代。

### 6.6 灵活性需求

用户进行注册时，能够辨别信息的合法性，如手机号是否满足标准格式，邮箱格式是否正确等。

### 6.7 扩展性需求

扩展性应包含开发与使用两个层面，对于开发方面：

1. 功能扩展：可在不影响原有系统使用的情况下迅速开发新进功能模块。
2. 平台扩展：可从单一服务器的节点扩展至服务器集群以增加响应效率，扩展过程中系统代码不需要发生变更。
3. 用户群体的扩展：可从在校大学生扩展到毕业从业人员。

### 6.8 数据管理能力需求

考虑到青年租房需求规模，本系统应至少能存储10000名用户的个人信息，并有能力快速存储与处理这些用户所拥有的需求、商品、交易与即时通讯信息。同时系统应能满足如下要求：

最高200人同时在线的提交请求：在这一程度的访问量下应保障所有访问均可以得到正确响应，响应时间不应超过500ms。

平均50人同时在线的提交请求：在这一程度的访问量下应保障所有访问均可以得到正确响应，响应时间不应超过300ms。

### 6.9 故障处理能力需求

考虑到本软件的使用场景比较复杂，尤其是其中设计数据库操作的部分，软件依然可能在运行过程中遭遇在开发、设计过程中未预料到的错误。因此，开发者应当实现以下要求，从而保证软件在发生错误时，维护人员能够及时地追踪、定位到错误的来源：

1. 集中保存异常日志，并至少从时间、操作内容和发起者三方面对操作与异常进行描述。例如，所有直接对数据库进行修改的操作都应在软件日志中写明操作发生的时间、具体的操作类型、操作的调用者等信息。
2. 在软件崩溃时，保存应用程序的内存空间dump文件。应用程序崩溃时的内存数据将能够极大地帮助开发人员检查程序中的错误，节约开发者与使用者之间的沟通成本。

## 7．目标系统界面

### 7.1界面需求

输入设备：键盘、鼠标

输出设备：显示器

显示风格：Web页面

显示方式：自动适配浏览器的长宽

输出格式：以网页的方式输出

### 7.2界面设计

这一部分仅为暂时的页面设计，即初稿，后期可能会做出相应的调整，最终的页面情况请参考软件设计说明书。

**7.2.1 用户登录页面**

该页面应实现如下功能：提供给已注册用户登录账户。页面中包含了填写账号、密码、登录按钮和登录身份，其中包括房客、房东、客服和师傅；当用户登录失败时，页面中会弹出相应的失败提示，方便用户修改登录信息。若登录成功，用户将直接跳到主页面。

**7.2.2 用户注册页面**

该页面应实现如下功能：用户注册。页面中包含了填写手机号或邮箱、密码和注册按钮，包含了注册为房客或房东的选项；当用户注册失败时，页面中会弹出相应的失败提示，方便用户修改信息重新注册。若注册成功，页面会提供填写个人详细信息的表单，以及确认按钮，用户可以选择填写或直接点击确定按钮完成注册。注册成功后将直接登录并跳转到主页面。

**7.2.3用户租房页面**

该页面应实现如下功能：选择长租或短租、查询并选择租房房间、提交租房申请、提交租金或导出订单、长租用户续约等。页面中主要含有租房选项、多种可供用户选择的房间。点击条目会跳转到主页面进行选择房间，并根据选择的租房类型跳转到最终步骤。

**7.2.4用户查询页面**

该页面应实现如下功能：查询订单内容、进行长租续约、对订单进行相关投诉。页面中主要含有用户订单的相关内容，包括房间信息，到期时间等，以及进行长租续约的选项。点击续约条目会跳转到用户租房页面进行续约、点击投诉条目可以进行相关投诉，可以对投诉处理结果进行评价。

**7.2.5客服管理租客页面**

该页面应实现如下功能：查询并修改租客资料。页面中主要含有所有租客的相关资料，以及查询选项。客服可以通过输入租客注册时部分资料查找到租客全部信息并进行修改操作。

**7.2.6客服管理合同页面**

该页面应实现如下功能：查询并修改合同资料。页面中主要含有所有合同的相关资料，以及查询选项。客服可以通过输入租客查找到合同全部信息并进行管理。

**7.2.7客服管理房间页面**

该页面应实现如下功能：查询房间信息并进行相关操作。页面中主要含有所有房间的相关资料，以及查询修改选项。客服可以通过输入房间号查找到房间信息，并可对房间信息进行增添以及暂停出租等操作。

**7.2.8客服管理工单页面**

该页面应实现如下功能：建立工单、安排师傅、回复投诉。页面中主要含有用户所提交的投诉。客服可以通过进入相关投诉，对投诉进行回复，并为投诉内容建立工单安排师傅处理。

**7.2.9师傅处理工单页面**

该页面应实现如下功能：接收工单、回复工单。页面中主要含有由客服审核并导入的用户所提交的投诉内容，师傅对工单进行接收处理并回复。

**7.2.10房东页面**

该页面应实现如下功能：上传房间信息。页面中主要用途为房东提交房源信息。其中包括房间基础介绍，大小，地理环境，周边环境，租金等信息，以及若干张房间图片介绍，并提供相应的房产证明。提交成功后相应房源会被上传至主页面房源信息之中。

**7.2.11主页面**

该页面应实现如下功能：用户注册和登录。页面中主要含有房间基本信息、房东头像和登录和注册按钮。用户点击房间后可以跳转到相应房间的浏览界面，在搜索框输入信息点击搜索按钮后会跳转到搜索详情页面，点击头像会跳转到个人空间页面，点击登录和注册按钮会跳转到登录页或者注册页。