**文档编号：旅游租房交易系统*– 1.0***

***旅游租房交易系统***

**软件设计规格说明书**

**日期：2020年10月14日**

**文档变更历史记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 变更日期 | 变更人员 | 变更内容详情描述 | 版本 |
| 1 | 2020\10\14 | 全员 | 撰写了软件设计规格说明书初稿 | V1.0 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**目录**

[1、引言 4](#_Toc18346)

[1.1 编写目的 4](#_Toc31179)

[1.2 读者对象 4](#_Toc251)

[1.3 软件项目概述 4](#_Toc13860)

[1.4 文档概述 4](#_Toc3116)

[1.5 定义 4](#_Toc26748)

[1.6 参考资料 4](#_Toc15237)

[2、软件设计约束 4](#_Toc16618)

[2.1 软件设计目标和原则 5](#_Toc20973)

[2.2 软件设计的约束和限制 5](#_Toc9534)

[3、软件设计 5](#_Toc6764)

[3.1 软件体系结构设计 5](#_Toc9972)

[3.2 用户界面设计 6](#_Toc20695)

[3.3 用例设计 6](#_Toc27048)

[3.4 类设计 6](#_Toc3319)

[3.5 数据设计 6](#_Toc1592)

[3.6 部署设计 6](#_Toc11160)

# **1、引言**

## 1.1 编写目的

软件设计过程包括的体系结构设计、用户界面设计、用例设计、子系统/构件设计、类设计和数据设计分别从不同的层次（从宏观到微观、从全局到局部）、不同的视角（从结构到行为、从模块到数据）对软件系统进行了设计，产生了不用的软件制品（如体系结构模型、用例实现模型、用户界面模型、子系统/构件模型、数据设计模型、部署模型等等）。在完成上述所有设计工作之后，需要将这些软件设计成果进行整合，形成一个系统、完整的软件设计方案。本文档以软件设计规格说明书的形式描述了该设计方案，方便评审人员对设计方案的正确性、合理性等方面进行评审。

## 1.2 读者对象

用户、软件设计人员、程序员、软件需求分析人员、质量保证人员、软件测试工程师、配置管理工程师。

## 1.3 软件项目概述

* 项目名称: 基于网页版的旅游租房交易系统
* 用户单位: 中国地区内所有旅游出行的游客
* 开发单位: 北京信息科技大学软工同路人团队
* 软件项目的背景和大致功能：

旅游租房交易平台是为了满足市场经济的快速发展和全国各地的游客对于房屋租赁的需求，利用先进的管理手段来提高房屋租赁的管理水平，提高游客旅游的便利性。本产品可以使得房客和房东之间的房屋出售和租赁更加便利，房东可以将自己出租的房屋放在平台上展示，并且添加出租房屋的主要信息，如：房屋价格、位置、平米数等，并租赁自己满意的房屋，也可以在退房后提出自己的评价。

正在开发的项目为旅游租房交易平台，本项目的提出者为北京信息科技大学软工同路人团队，用户为中国地区内所有旅游出行的游客，为他们提供旅游各地遇到的各种优质房源，简化交易过程，让用户在旅游过程中能省心交易、舒适居住，目前为独立平台，与现行其他系统无接口。

## 1.4 文档概述

1）软件的设计约束部分。它包括软件设计目标和原则、软件设计受到的约束和限制。

2）软件的设计部分。它主要分为软件体系结构设计、用户界面设计、用例设计、类设计、数据设计以及部署设计。

## 1.5 定义

无

## 1.6 参考资料

[1].软件工程.齐治昌，谭庆平，宁洪.北京:高等教育出版社

[2].需求分析与设计.马素霞译.北京:机械工业出版社

[3].面向使用的软件设计.刘正捷译.北京:机械工业出版社

[4].xxx项目软件设计规格说明书

# **2、软件设计约束**

## 2.1 软件设计目标和原则

软件设计的目标是，根据软件系统的需求（包括功能性需求和非功能性需求），综合考虑软件开发过程中的各种制约因素（如技术、资源、进度等），遵循软件工程的设计原则（如模块化、信息隐藏、问题分解等），给出软件系统的实现解决方案和蓝图，产生可指导编码实现的设计模型及文档。

软件设计活动还须遵循相关的策略和原则，以指导软件设计人员的行为，并对设计成果提出约束和要求。具体的，这些设计策略和原则描述如下。

1. 抽象和逐步求精的原则
2. 模块化与高内聚度、低耦合度的原则
3. 信息隐藏的原则
4. 多视点以及关注点分离的原则
5. 软件重用的原则
6. 迭代设计的原则
7. 可追踪性的原则

## 2.2 软件设计的约束和限制

* 运行环境要求：windows系统
* 开发语言：Java
* 标准规范：Java编码风格（规范）
* 开发工具：IDEA、VUE

# **3、软件设计**

## 3.1 软件体系结构设计

分为前端用户浏览界面和后台管理系统

1、 注册登录功能：提供注册功能，用户需要登录之后才能进行下单操作；提供密码的找回功能。

2、 浏览功能：在网站主页提供热门房子的浏览，显示房间的大致信息，如标题、评价等；可根据热门地点进行浏览；点击房子可进入查看房间详细信息，如是否有wifi、多少间卧室、能够住几人。

3、 搜索功能：用户可根据房间名称或者地点进行房间的查询。

4、 下单功能：用户输入真实姓名、身份证号、手机号、入住时间退房时间、入住人数即可进行房间预定，进入支付过程。用户支付成功后，房子被锁定，无法被其他用户所查看和预定。

5、 支付功能：用户下单后，提供支付宝/微信的付款方式。

6、 退房功能：用户下单后，可在指定之间内进行订单的取消和申请更改功能。

7、 发布功能：用户可申请成为出租人，出租自己的空闲房子，但需要提供必要的信息，然后进入审核过程，审核通过房子即可出租。

8、 用户功能：用户可更改自己的信息，如头像、用户、手机号等。用户可查看自己的订单。

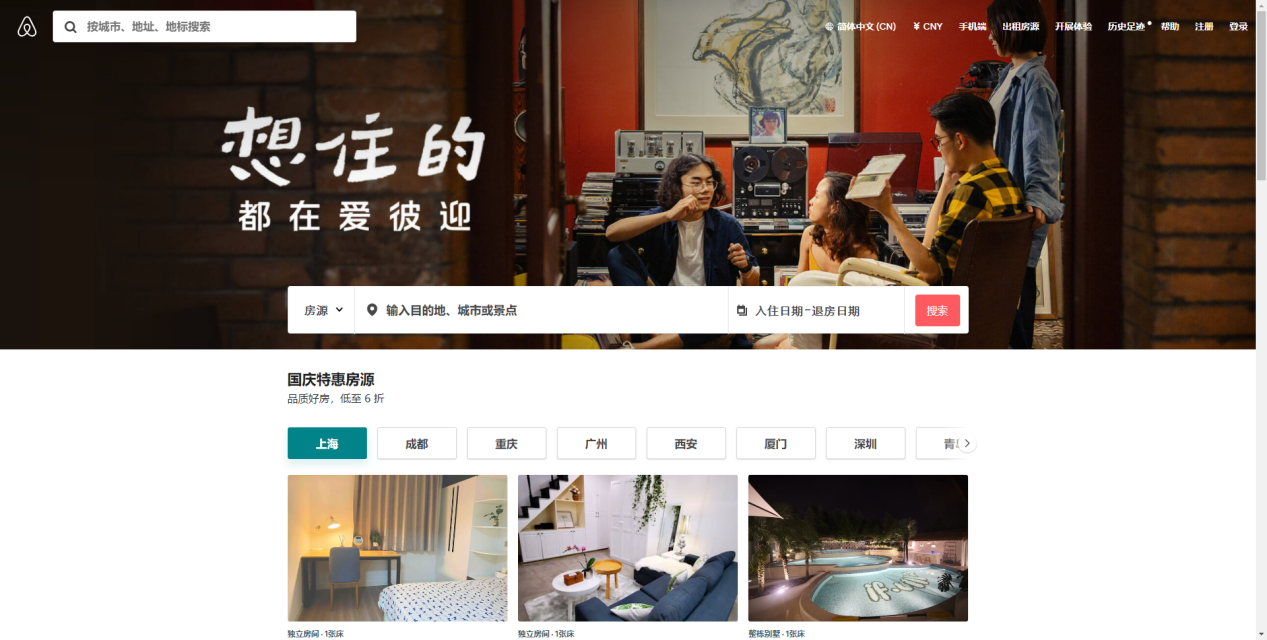
9、 审核功能：管理员登录后端管理系统，可进行房子的发布审核；以及查看网站的访问量统计。

10、 评论功能：用户在订单完成后可对该房子进行评价，满分五颗星并可进行详细描述供其他用户查看。用户评论后可进行评论删除。

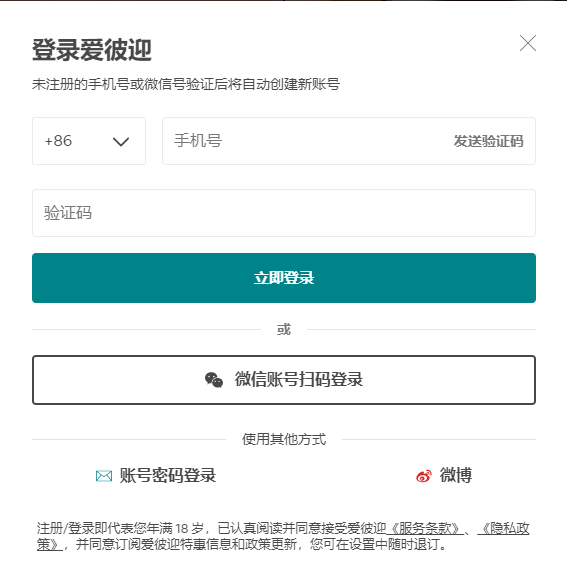
## 3.2 用户界面设计

项目模板：

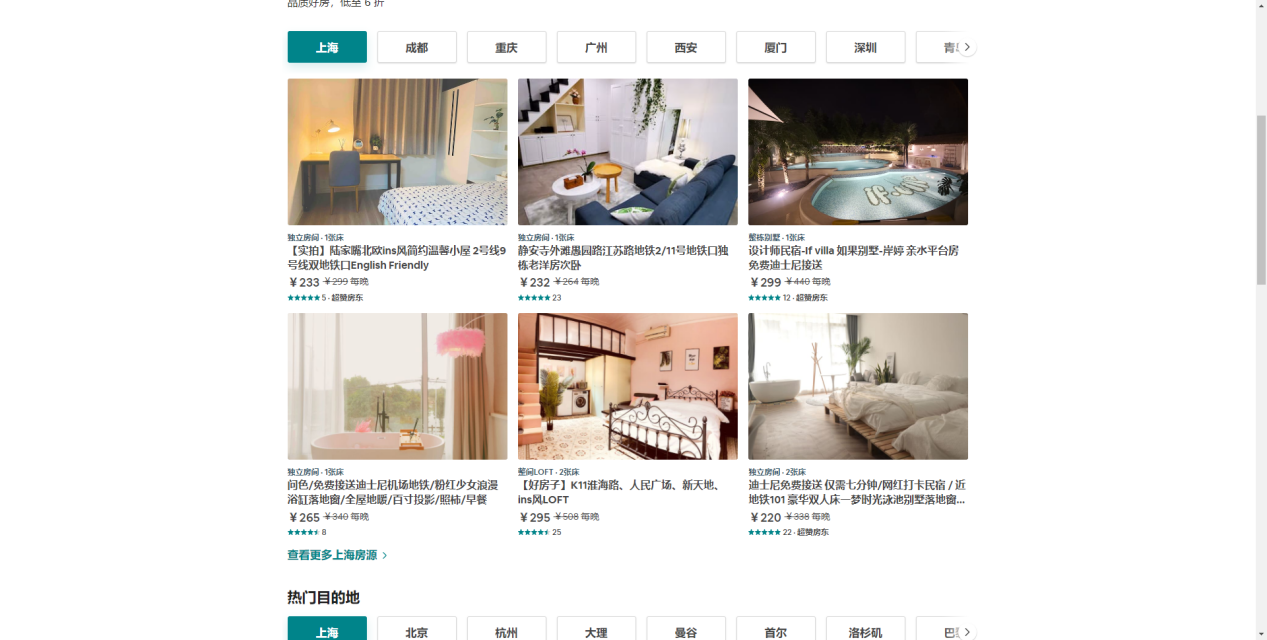
首页



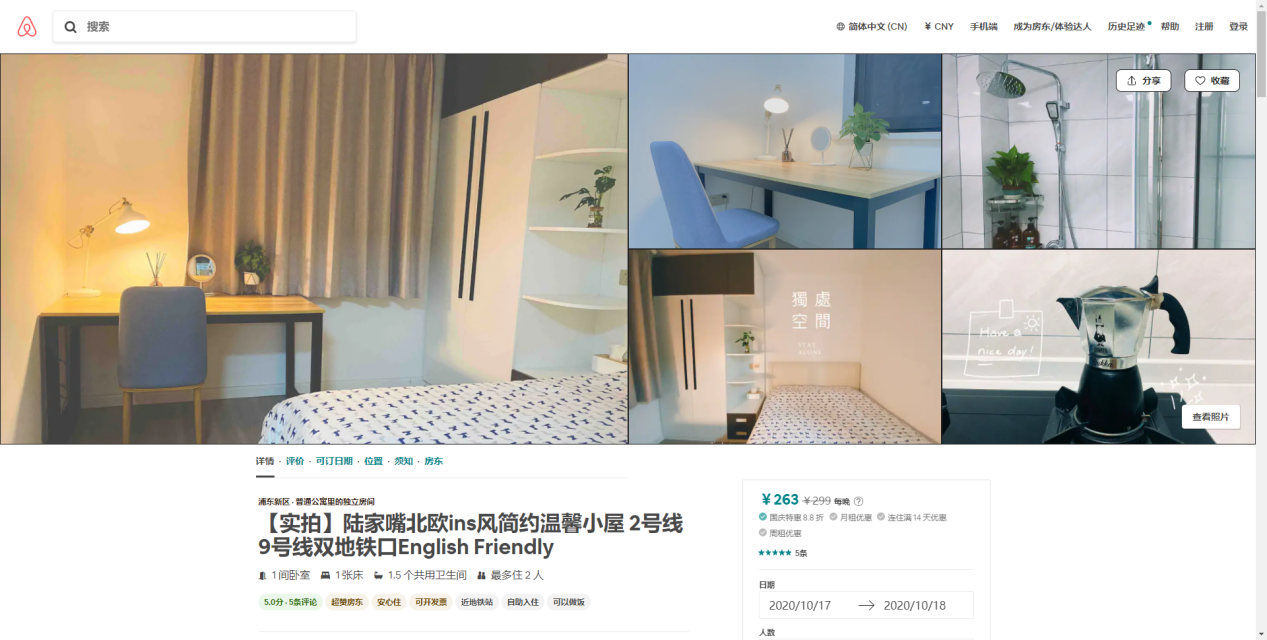
登录/注册界面：

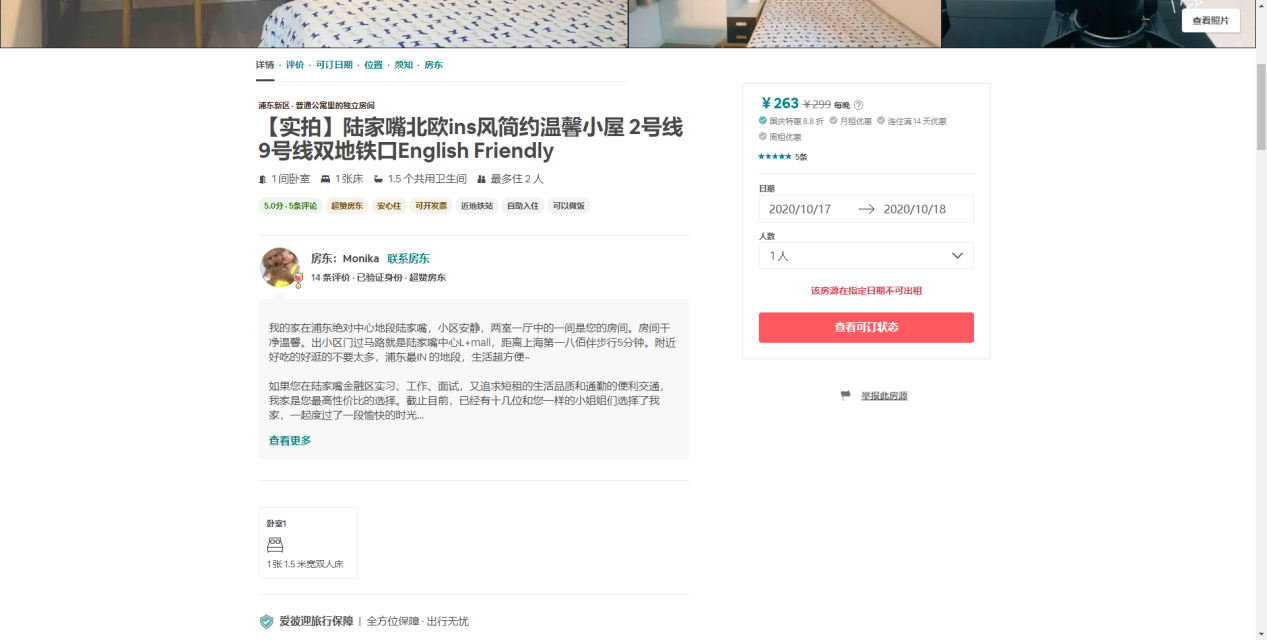


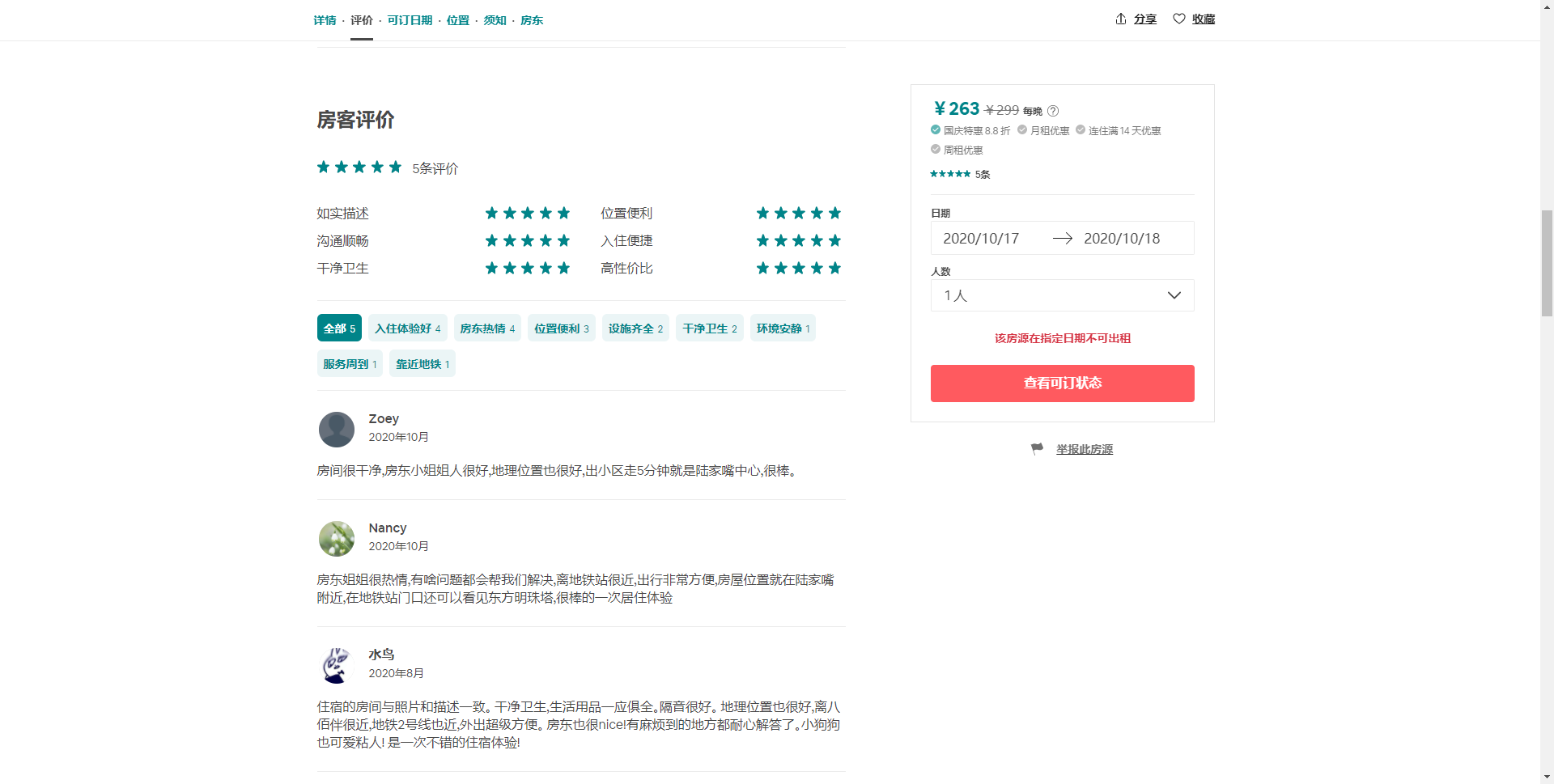
浏览功能：



房屋详情/订购界面：



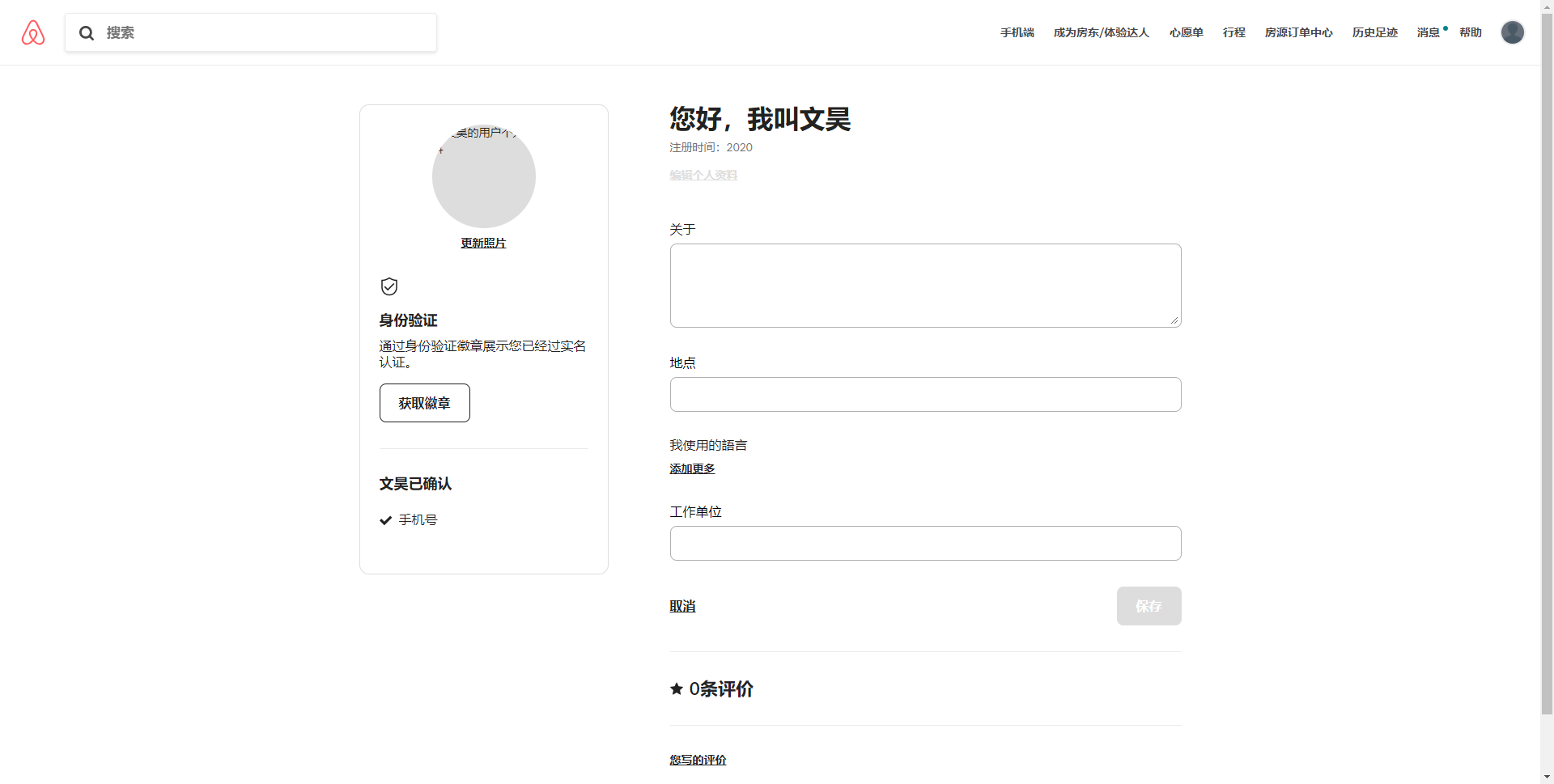


房屋评价界面：

发布房屋界面：



个人资料界面：

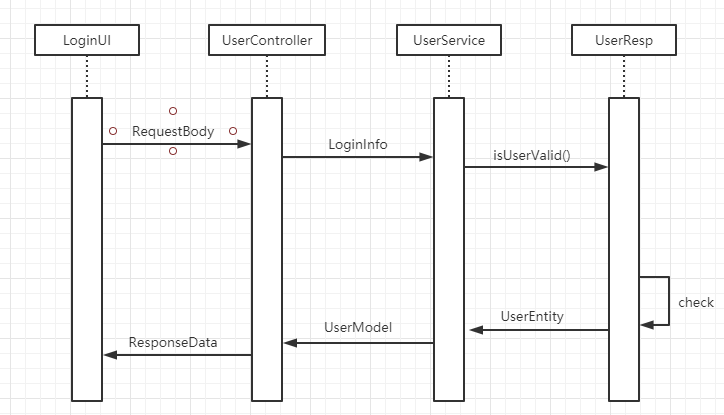


## 3.3 用例设计

（1）“用户登录”用例实现的设计方案

“用户登录”功能的实现主要是通过“UserLibrary”对象提供的服务，查询数据库中是否有用户输入的账号和密码信息，从而来判定该用户的身份是否合法。

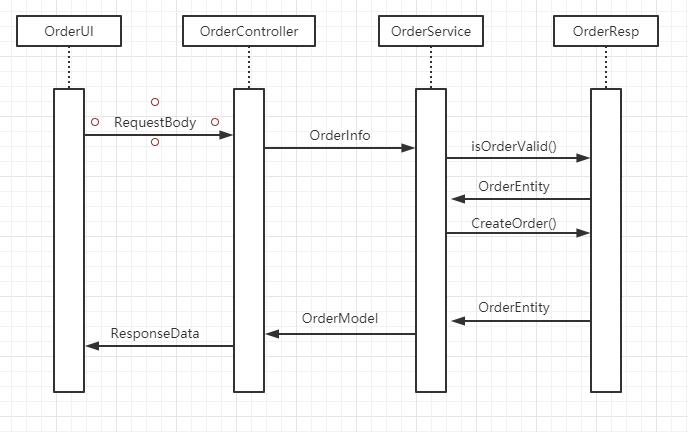
首先，用户通过边界类“LoginUI”对象输入登录的账户和密码，随后该对象向控制类“UserController”对象发消息“login(account, password)”，以请求登录到系统之中。接收到消息后，“UserController”通过实体类“LoginInfo”对象向“UserService”发消息“verifyUserValidity(account, password)”以验证用户提交的账号和密码是否合法。通过方法“isUserValid(account, password)”来判断用户身份的合法性，并将验证的结果“VerificationResult”返回给“UserController”对象，“UserController”对象以此进一步地将登录成功与否的消息发送给“LoginUI”对象。一旦登录成功，系统将界面重定向到“IndexUI”主界面。



（2）“订购房屋”用例实现的设计方案

“订购房屋”具体描述如下：用户通过浏览房间，选择心仪的房间查看相关信息并选择下单订购，在订购界面填写相关的信息，如手机，真实姓名，身份证号，入住人数及其相关信息等，然后点击下单，支付成功后生成订单。

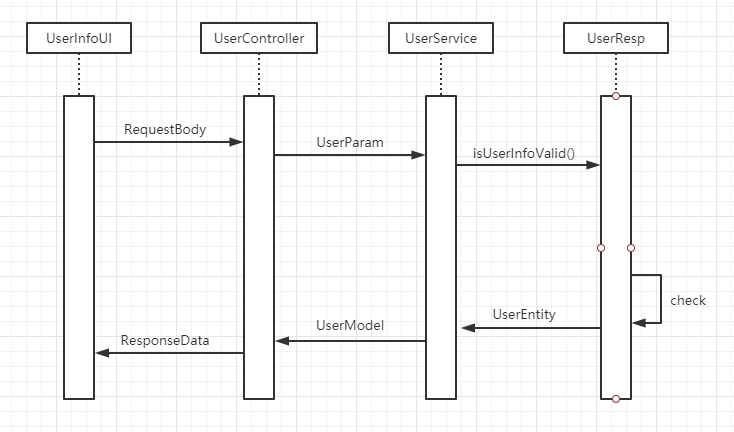
功能实现主要通过边界类“OrderUI”对象获取相应所需要的信息，然后向控制类OrderController发送消息。接收到消息后，OrderController通过实体类OrderLibrary将消息发送至OrderService，通isOrderValid(orderLibrary)验证订单是否合法，如果合法，则交由createOrder()创建订单，创建成功后将订单相关信息返回给OrderUI，系统将界面定位到订单支付界面。



1. “修改资料”用例实现的设计方案

“修改资料”实现描述如下：用户从主页进入个人信息界面，点击编辑进行个人信息的编辑，更改需要更改的信息，如头像、手机、名称等，更改完成后点击保存，系统发送更改的相关信息至后端。

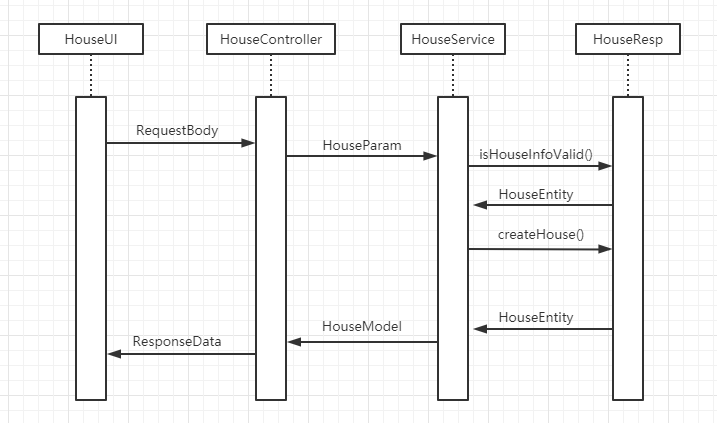
功能实现主要通过边界类“UserUI”对象获取相应所需要的信息，然后向控制类UserController发送消息。接受到消息后UserController通过实体类UserInfo将消息发送给UserService，通过modifyUserInfo（UserInfo）修改用户信息，最后将修改成功与否的消息返回，由前端提示相关信息。



1. “发布房屋”用例实现的设计方案

“发布房屋”实现描述如下：用户从主页的发布房屋进入发布房屋界面，填写相关房屋的信息，如标题，描述，价格等，填写完成后提交至后台，此时尚未发布。需要管理员在后台审核通过之后才正式上线。

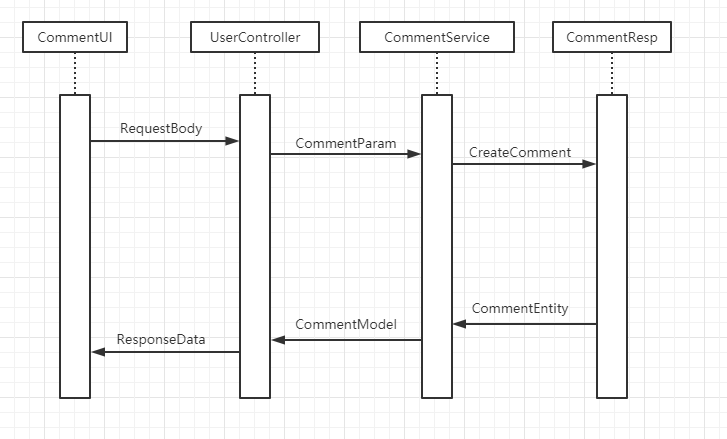
功能实现主要通过边界类“HousePublishUI”对象获取相应所需要的信息，然后向控制类HouseController发送消息。接收到消息后，HouseController通过实体类HouseLibrary将消息发送至HouseService，通isHouseValid(houseLibrary)验证房屋是否合法，如果合法，则交由createHouse()创建房屋，创建成功后将成功与否信息返回，并将房屋相关信息返回给管理员，管理员通过验证，系统将界面定位回界面。



1. “发布评论”用例实现的设计方案

“发布评论”实现描述如下：用户在完成订单之后，可以对商家进行评价，仅能评价一次，不超过200字，可以附加图片若干。用户填写完评论，选择对应的评价星级，提交成功后即评论成功，可及时显示相关评论；但此评论同样会被提交到后台给管理员审核，如若管理员审核不通过，有权将该不合规的评论进行删除操作，

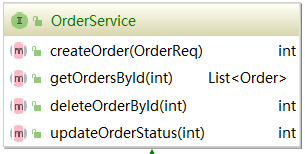
功能实现主要通过边界类“CommentUI”对象获取相应所需要的信息，然后向控制类UserController发送消息。接收到消息后，UserController通过实体类CommentLibrary将消息发送至UserService，通isCommentValid(commentLibrary)验证评论是否合法，如果合法，则交由createComment()创建房屋，创建成功后将成功与否信息返回，系统将界面定位回界面。

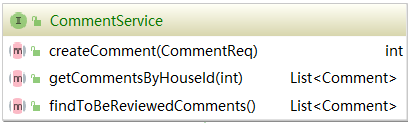


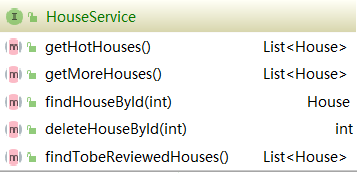
## 、3.4 类设计

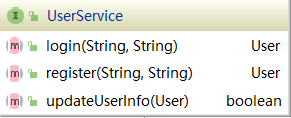
|  |  |
| --- | --- |
| Controller | |
| 类名 | 说明 |
| UserController | 处理关于用户的请求，如注册、登录、信息修改等，进行相关处理，具体与数据库相关事务交由Service层处理 |
| OrderController | 处理关于订单的请求，如下订单、订单信息修改、订单取消等，进行相关处理，具体与数据库相关事务交由Service层处理 |
| HouseController | 处理关于房屋的请求，如发布房屋、修改房屋信息、获取房屋信息等，进行相关处理，具体与数据库相关事务交由Service层处理 |

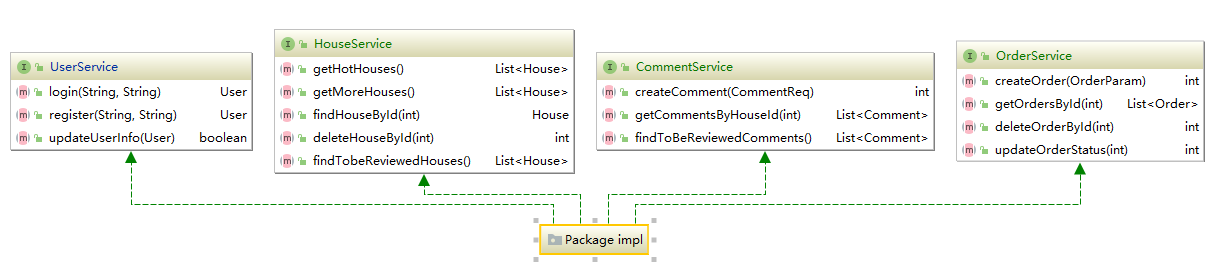
|  |  |
| --- | --- |
| Service | |
| 类名 | 说明 |
| UserService | 处理由Controller层发送的相关请求，并且调用相关的jpa进行数据库的操作。 |
| OrderService | 处理由Controller层发送的相关请求，并且调用相关的jpa进行数据库的操作。 |
| HouseService | 处理由Controller层发送的相关请求，并且调用相关的jpa进行数据库的操作。 |
| CommentService | 处理由Controller层发送的相关请求，并且调用相关的jpa进行数据库的操作。 |











## 3.5 数据设计

（1）设计永久保存数据的数据库表及字段

|  |  |  |
| --- | --- | --- |
| tbl\_user | | |
| id | int | 主键，自增 |
| name | nchar(10) | 用户名 |
| email | varchar(50) | 邮箱，用于登录 |
| password | char(20) | 账户密码 |
| phone | char(13) | 用户手机号 |
| is\_ad | tinyint(1) | 是否为管理员 |
| head\_portrait | char(100) | 头像 |
| create\_on | datetime | 创建时间 |
| modified\_on | datetime | 修改时间 |

|  |  |  |
| --- | --- | --- |
| tbl\_house | | |
| id | int | 主键，自增 |
| owner\_id | int | 房屋所属用户的id |
| title | nvarchar(40) | 房屋标题 |
| description | nvarchar(200) | 房屋描述 |
| price | int | 房屋价格 |
| photo | char(100) | 房屋描述照片 |
| bedroom\_num | tinyint(1) | 卧室数量 |
| bathroom\_num | tinyint(1) | 卫生间数量 |
| location | nchar(50) | 地理位置 |
| max\_ppl | tinyint(1) | 最多住几人 |
| status | tinyint(1) | 房屋状态，是否审核通过 |
| create\_on | datetime | 创建时间 |
| modified\_on | datetime | 修改时间 |

|  |  |  |
| --- | --- | --- |
| tbl\_order | | |
| id | int | 主键，自增 |
| buyer\_id | int | 买家id |
| seller\_id | int | 卖家id |
| house\_id | int | 房屋id |
| price | int | 订单价格 |
| checkin\_date | datetime | 入住时间 |
| checkout\_date | datetime | 退房时间 |
| ppl\_info | int | 入住用户的相关信息 |
| status | tinyint(1) | 订单状态 |
| create\_on | datetime | 创建时间 |
| modified\_on | datetime | 修改时间 |

|  |  |  |
| --- | --- | --- |
| tbl\_comment | | |
| id | int | 主键，自增 |
| house\_id | int | 房屋id |
| critic\_id | int | 评论者id |
| content | nvarchar(100) | 评论内容 |
| rate | tinyint(1) | 评价等级 |
| create\_on | datetime | 创建时间 |
| modified\_on | datetime | 修改时间 |

## 3.6 接口设计



