酒店管理系统

HMS (Hotel Management System)

体系结构设计文档

V2.0

钛4+小组 (Titanium Quartet)

蒋文荟 李佩瑶 李珍鸿 李一然

2016/10/11

目录

[更新历史 3](#_Toc463991677)

[一、软件体系结构逻辑设计方案 4](#_Toc463991678)

[二、开发包设计 5](#_Toc463991679)

[三、开发包图 8](#_Toc463991680)

[3.1 酒店管理系统客户端开发包图 8](#_Toc463991681)

[3.2 酒店管理系统服务器开发包图 9](#_Toc463991682)

[3.3 酒店管理系统用户界面跳转示意图 10](#_Toc463991683)

# 更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 变更原因 | 版本号 |
| 李一然 | 2016/10/10 | 编写逻辑模型示意图和客户端及服务器包图 | V1.0 |
| 李珍鸿 | 2016/10/10 | 增加开发包设计表格 | V1.1 |
| 李佩瑶 | 2016/10/10 | 增加界面跳转示意图 | V1.2 |
| 李珍鸿 | 2016/10/13 | 编写部分逻辑层模块的接口规范 | V2.0 |
| 李佩瑶 | 2016/10/13 | 编写部分逻辑层模块的接口规范 | V2.1 |
| 蒋文荟 | 2016/10/14 | 增加数据库表 | V2.2 |
| 李一然 | 2016/10/14 | 修改逻辑层模块的接口规范 | V2.3 |
| 李珍鸿 | 2016/10/14 | 编写引言、产品概述、逻辑视角、组合视角及用户界面层的分解 | V3.0 |
| 蒋文荟 | 2016/10/15 | 修改数据库表，增加数据层的分解 | V3.1 |

# 引言

## 1.1编制目的

本报告详细完成对酒店管理系统的概要设计，达到知道详细设计和开发的目的，同时实现和测试人员及用户的沟通。

本报告面向开发人员，测试人员以及最终用户而编写，是了解系统的导航

## 1.2词汇表

|  |  |  |
| --- | --- | --- |
| 词汇名称 | 词汇含义 | 备注 |
| HMS | 酒店管理系统 |  |
| ui | 用户界面 |  |
| bl | 业务逻辑 |  |
| RMI | 远程方法调用 |  |

# 二，产品概述

参考酒店管理系统用例文档和酒店管理系统软件需求规格说明中对产品的描述。

# 三，逻辑视角

酒店管理系统中，选择了分层体系结构风格，将系统分为3层（界面层，业务逻辑层，数据层）能够很好地展示整个高层抽象。界面层包含图形用户界面的实现，业务逻辑层包含业务逻辑处理的实现，数据层负责数据的持久化和访问。分层体系结构的逻辑视角和逻辑设计方案如图一和图二所示：



图1 参照体系结构风格的包图表达逻辑视角



图2 软件体系结构逻辑设计方案

# 四，组合视角

## 4.1开发包图

酒店管理系统的最终开发包设计如表1所示。

表1 酒店管理系统的最终开发包设计

|  |  |
| --- | --- |
| 开发包 | 依赖的其他开发包 |
| mainui | usermainui, personalui, createorderui, strategyui, hotelinfoui, searchhotelui, manageroomui, manageorderui, managehotelui, manageuserui |
| usermainui | usermainblservice , vo, 界面类库包 |
| personalui | personalblservice, vo, 界面类库包 |
| createorderui | createorderblservice, vo, 界面类库包 |
| strategyui | strategyblservice, vo, 界面类库包 |
| hotelinfoui | hotelinfoblservice, vo, 界面类库包 |
| searchhetelui | searchhetelblservice, vo, 界面类库包 |
| manageroomui | manageroomblservice, vo, 界面类库包 |
| manageorderui | manageorderblservice, vo, 界面类库包 |
| managehotelui | managehotelblservice, vo, 界面类库包 |
| manageuserui | manageuserblservice, vo, 界面类库包 |
| usermainblservice |  |
| usermainbl | userdataservice, po |
| personalblservice |  |
| personalbl | userdataservice, creditdataservice, po |
| createorderblservice |  |
| createorderbl | orderdataservice, roomdataservice, creditdataservice, strategydataservice, po |
| strategyblservice |  |
| strategybl | mainuserbl, strategydataservice, locationdataservice, po |
| Hotelinfoblservice |  |
| hotelinfobl | manageorderbl , roomdataservice, strategydataservice, hotelinfodataservice, reviewdataservice, po |
| searchhotelblservice |  |
| searchhotelbl | manageorderbl, roomdataservice, reviewdataservice, locationdataservice, hotelinfodataservice, po |
| manageroomblservice |  |
| manageroombl | mainuserbl, orderdataservice, roomdataservice, po |
| manageorderblservice |  |
| manageorderbl | mainuserbl, orderdataservice, roomdataservice, creditdataservice, po |
| managehotelblservice |  |
| managehotelbl | mainuserbl, userdataservice, hotelinfodataservice, po |
| manageuserblservice |  |
| mangeuserbl | creditdataservice, userdataservice |
| orderdataservice | Java RMI, po |
| orderdata | databaseutility, po |
| roomdataservice | Java RMI, po |
| roomdata | databaseutility, po |
| userdataservice | Java RMI, po |
| userdata | databaseutility, po |
| creditdataservice | Java RMI, po |
| creditdata | databaseutility, po |
| strategydataservice | Java RMI, po |
| strategydata | databaseutility, po |
| hotelinfodataservice | Java RMI, po |
| hoteldatainfo | databaseity, po |
| reviewdataservice | Java RMI, po |
| reviewdata | databaseutility, po |
| Locationdataservice | Java RMI, po |
| locationdata | databaseutility, po |
| po |  |
| vo |  |
| 界面类库包 |  |
| Java RMI |  |
| databaseutility | JDBC |

## 图3 酒店管理系统客户端开发包图

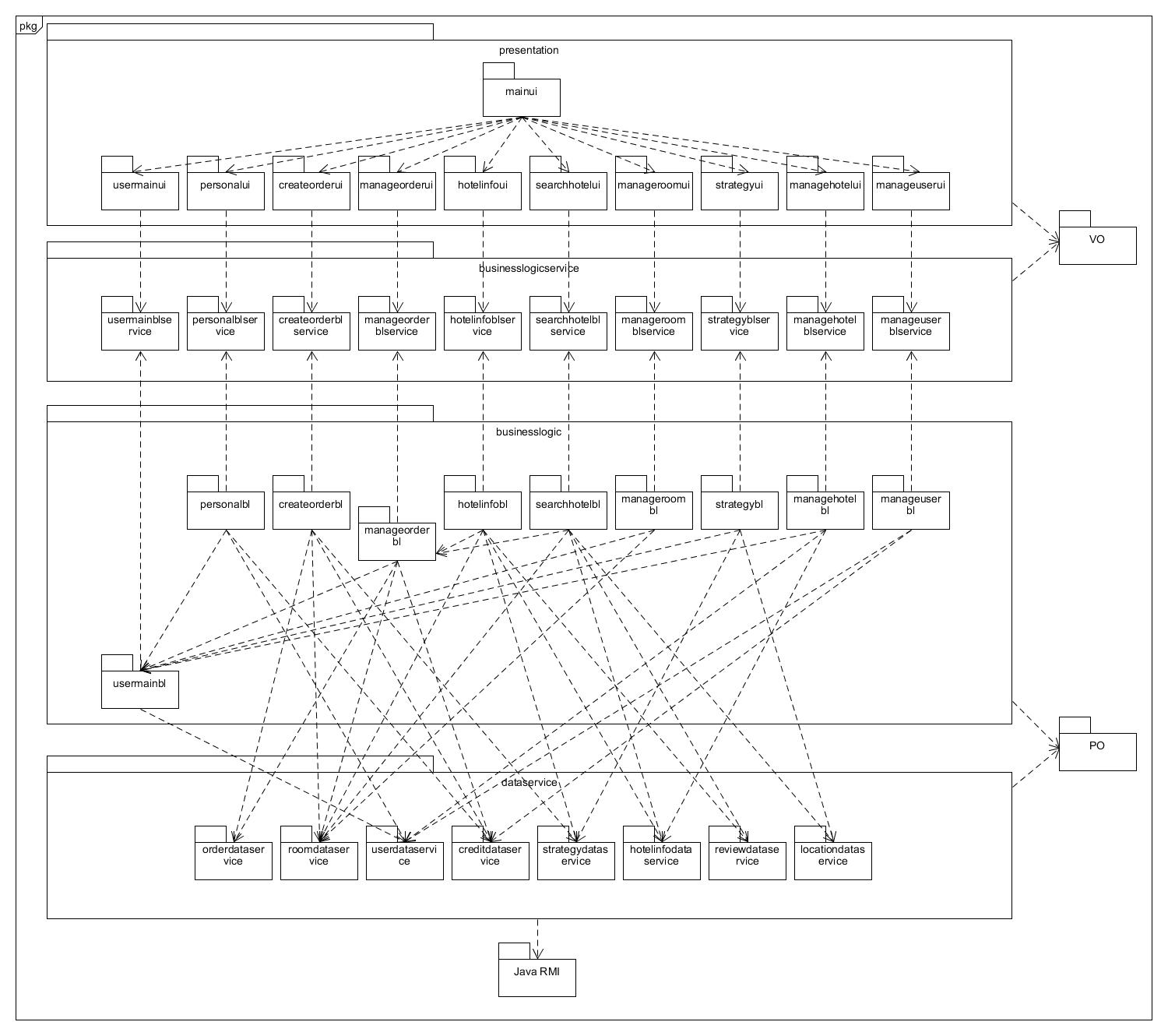


图4 酒店管理系统服务器端开发包图

## 4.2 运行时进程

在酒店管理系统中，会有多个客户端进程和一个服务器端进程，其进程图如图五所示，结合部署图，客户端进程是在客户端机器上运行，服务器进程在服务器端机器上运行

图5 进程图

## 4.3 物理部署

酒店管理系统中客户端构件是放在客户端机器上，服务器端构件是放在服务器端机器上。在客户端节点上，还要部署RMIStub构件。由于java RMI构件属于JDK 8.0的一部分。所以，在系统JDK环境已经设计好的情况下，不需要再独立部署。部署图如图六所示。



图6 部署图

# 五，接口模块

## 5.1 模块的职责

客户端模块和服务器模块视图分别如图7和图8所示。客户端各层和服务器各层的职责分别如表2和表3所示。



图7 客户端模块视图



图8 服务器端模块视图

表2 客户端各层的职责

|  |  |
| --- | --- |
| 层 | 职责 |
| 启动模块 | 负责初始化网络通信机制，启动用户界面。 |
| 用户界面层 | 基于窗口的进销存系统客户端用户界面。 |
| 业务逻辑层 | 对于用户界面的输入响应和业务处理逻辑。 |
| 客户端网络模块 | 利用Java RMI机制查找RMI服务,检测网络连接状态，进行断线重连 |

表3 服务器端各层的职责

|  |  |
| --- | --- |
| 层 | 职责 |
| 启动模块 | 负责初始化网络通信机制，启动用户界面。 |
| 数据层 | 负责数据的持久化及数据访问接口。 |
| 服务器端网络模块 | 利用Java RMI机制开启RMI服务，注册RMI服务 |

每一层只是使用下方直接接触的层。层与层之间仅仅是通过接口的调用来完成的。层间调用的接口如表四所示

表4 层之间调用的接口

|  |  |  |
| --- | --- | --- |
| 接口 | 服务调用方 | 服务提供方 |
| usermainblservice  personalblservice  createorderblservice  strategyblservice  hotelinfoblservice  searchhotelblservice  manageroomblservice  manageorderblservice  managehotelblservice  manageuserblservice | 客户端展示层 | 客户端业务逻辑层 |
| orderdataservice  roomdataservice  userdataservice  creditdataservice  strategydataservice  hotelinfodataservice  reviewdataservice  locationdataservice | 客户端业务逻辑层 | 服务器端数据层 |

借用用户管理用例来说明层之间的调用，如图九所示。每一层之间都是由上层依赖了一个接口（需接口），而下层实现这个接口（供接口）。UserBLService提供了User界面所需要的所有业务逻辑功能UserDataService提供了对数据库的增、删、改、查等操作。这样的实现就大大降低了层与层之间的耦合。

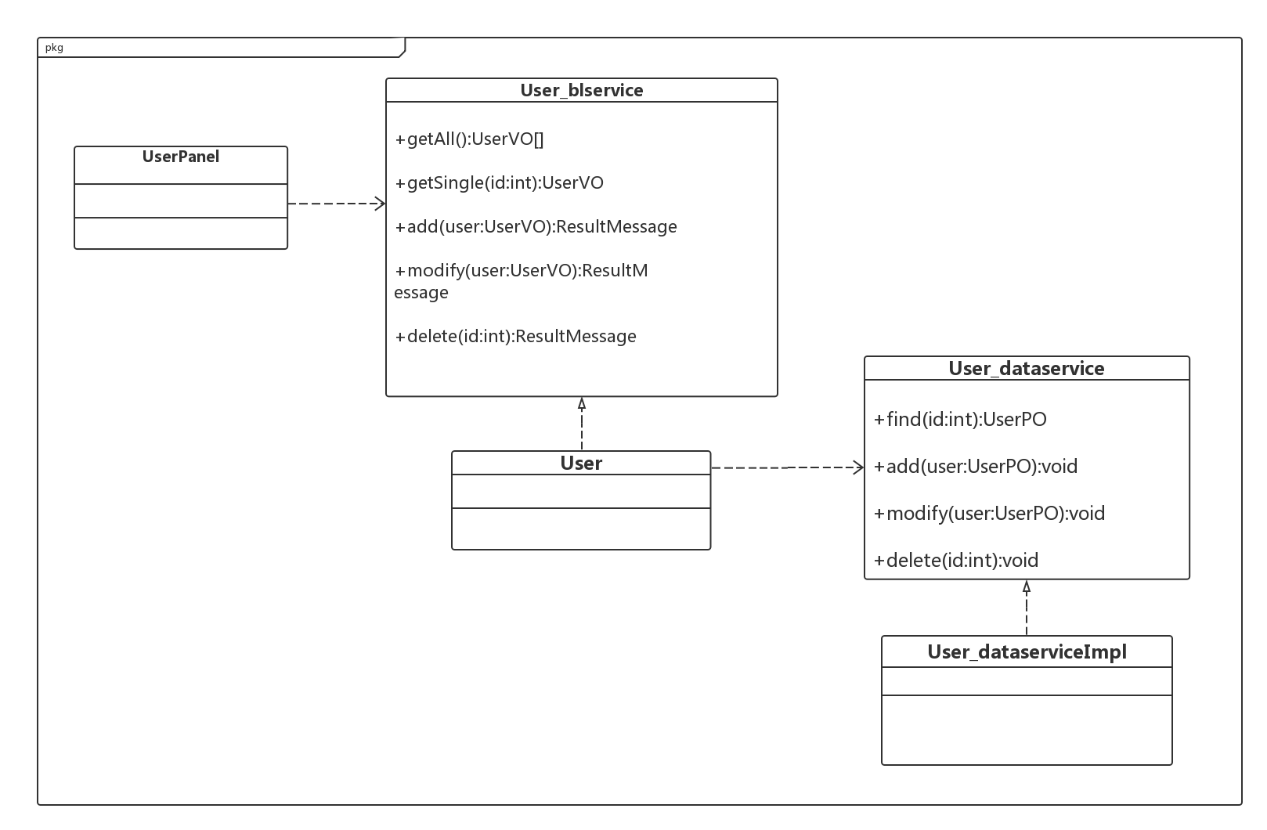


图9 用户管理用例之间调用的接口

## 5.2 用户界面层的分解

根据需求，系统存在45个用户界面。它们分别是：修改酒店工作人员信息界面，修改客户信息界面，修改网站营销人员信息界面，查看酒店工作人员信息界面，查看用户信息界面，查看网站营销人员信息界面，查找酒店工作人员界面，查找用户界面，查找网站营销人员界面，网站管理人员界面，添加酒店及其工作人员界面，登录界面，客户注册界面，酒店工作人员界面，查看酒店促销策略界面，查看可用客房界面，线下入住界面，修改酒店信息界面，执行订单界面，更新退房信息界面，浏览酒店订单界面，查看酒店基本信息界面，修改酒店基本信息界面，网站营销人员界面，查看未执行订单界面，撤销异常订单界面，查看网站促销策略界面，修改促销策略界面，会员折扣制度界面，商圈促销策略界面，特殊日期促销界面，信用充值界面，客户信息核对界面，查找客户界面，信用记录界面，客户界面，维护用户信息界面，注册会员界面，搜索酒店信息界面，酒店详情界面，生成订单界面，查看预定过的酒店界面，撤销未执行订单界面，浏览客户订单界面。

界面跳转如图10所示：

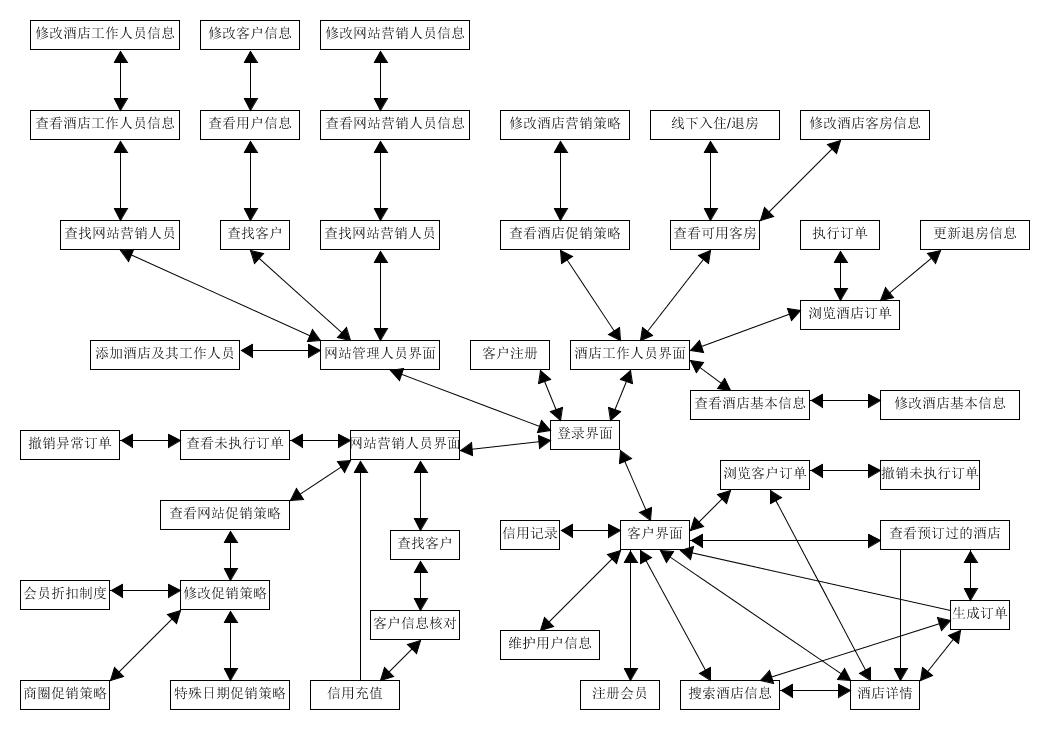


图10 用户界面跳转示意图

服务器端和客户端的用户界面设计接口是一致的，只是具体的页面不一样。用户界面类如图十一所示

图十一 用户界面类

### 5.2.1 用户界面层模块的职责

表5.2.1-1所示为用户界面层模块的职责

表5.2.1-1 用户界面层模块的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| MainFrame | 界面Frame，负责界面的显示和界面的跳转。 |

### 5.2.2 用户界面层的接口规范

如图5.2.2-1所示为用户界面层模块接口规范

|  |  |  |
| --- | --- | --- |
| usermainui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示Frame以及loginPanel |
| personalui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示用户信息界面 |
| createorderui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示createorderPanel |
| manageorderui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示manageorderPanel |
| hotelinfoui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示hotelinfoPanel |
| searchhotelui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示searchhotelPanel |
| managerooui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示manageroomPanel |
| strategyui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示strategyPanel |
| managehotelui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示managehotelPanel |
| manageuser\_ui | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示manageuserPanel |

用户界面层需要的服务接口如表5.2.2-2所示。

表5.2.2-2 用户界面层模块需要的服务接口

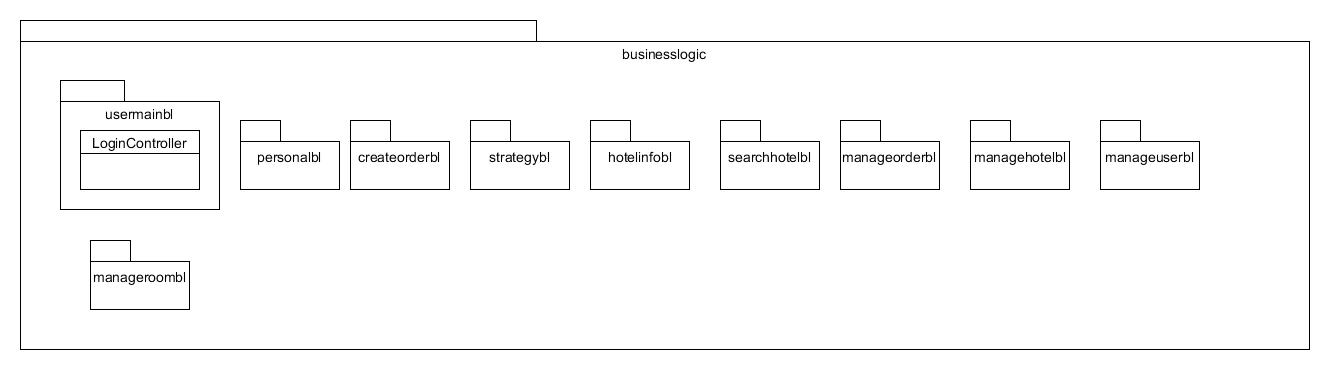
|  |  |
| --- | --- |
| 服务名 | 服务 |
| businesslogicservice.usermain\_blservice | 登陆界面的业务逻辑接口 |
| businesslogicservice.personal \_blservice | 处理用户信息的接口 |
| businesslogicservice.createorder\_blservice | 处理与生成订单相关的接口 |
| businesslogicservice.hotelinfoblservice | 处理各种酒店信息的接口 |
| businesslogicservice.manageorderblservice | 提供管理订单的接口 |
| businesslogicservice.strategyblservice | 提供促销策略的接口。 |
| Businesslogicservice.manageroomblservice | 负责管理房间的服务的接口 |
| businesslogicservice.manageuserblservice | 负责管理用户的接口 |
| businesslogicservice.managehotelblservice | 负责管理酒店的服务 |
| businesslogicservice.searchhotelblservice | 提供搜索酒店的接口 |
| businesslogicservice.managehotelblservice | 提供管理酒店的接口 |

### 5.2.3 用户界面模块设计原理

用户界面利用Java的Swing和AWT库来实现。

## 5.3业务逻辑层的分解

业务逻辑层包括多个针对界面的业务逻辑处理对象。例如，User对象负责处理登陆界面的业务逻辑；User对象负责处理管理用户的业务逻辑。业务逻辑层的设计如图5.3-1所示。



### 5.3.1 业务逻辑层的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| usermainbl | 提供验证用户登录的服务 |
| personalbl | 负责查看用户信息的服务 |
| createorderbl | 负责处理生成订单的服务 |
| managehotelbl | 负责管理酒店的服务 |
| Hotelinfobl | 负责处理酒店信息相关的服务 |
| manageorderbl | 负责管理订单服务。 |
| strategybl | 提供与促销策略相关服务。 |
| manageuserbl | 负责管理用户的服务 |
| manageroombl | 负责管理房间的服务 |
| searchhotelbl | 负责搜索酒店的服务 |

### 5.3.2业务逻辑层的接口规范