|  |
| --- |
| 互联网酒店管理系统  Hotel Property Management System |
| 软件体系结构描述文档 |
| V1.0 草稿 |

|  |
| --- |
| 焦立群、金灵益、黎奥邦、李恩铭  2016-10-11 |

更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 变更原因 | 版本号 |
| 焦立群 | 2016-10-11 | 最初草稿 | V1.0 |

目录

[更新历史 1](#_Toc463959294)

[1. 引言 3](#_Toc463959295)

[1.1 编制目的 4](#_Toc463959296)

[1.2 词汇表 4](#_Toc463959297)

[1.3 参考资料 4](#_Toc463959298)

[2. 产品概述 3](#_Toc463959299)

[3. 逻辑视角 3](#_Toc463959300)

[4. 组合视角 3](#_Toc463959301)

[4.1 开发包图 3](#_Toc463959302)

[4.2 运行时进程 3](#_Toc463959303)

[4.3 物理部署 3](#_Toc463959304)

[5. 接口视角 3](#_Toc463959305)

[5.1 模块的职责 3](#_Toc463959306)

[5.2 用户界面层的分解 3](#_Toc463959307)

[5.3 业务逻辑层的分解 3](#_Toc463959308)

[5.4 数据层的分解 3](#_Toc463959309)

[6. 信息视角 3](#_Toc463959310)

[6.1 数据持久化对象 3](#_Toc463959311)

[6.2 Txt持久化格式 3](#_Toc463959312)

[6.3 数据库表 4](#_Toc463959313)

1. 引言
   1. 编制目的

本报告详细完成对互联网酒店管理系统的概要设计，达到指导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及最终用户编写，是了解系统的导航。

* 1. 词汇表

|  |  |  |
| --- | --- | --- |
| 词汇名称 | 词汇含义 | 备注 |
| HPMS | 互联网酒店管理系统 | Hotel Property Management System |

* 1. 参考资料

互联网酒店管理系统HPMS软件需求规格说明\_V1.0正式版

互联网酒店管理系统HPMS用例文档\_V1.1正式版

1. 产品概述

参考互联网酒店管理系统用例文档和互联网酒店管理系统软件需求规格说明中对产品的概括描述。

1. 逻辑视角
2. 组合视角
   1. 开发包图
   2. 运行时进程
   3. 物理部署
3. 接口视角
   1. 模块的职责

客户端模块和服务器端模块视图分别如图和图所示。客户端各层和服务器端各层的职责分别如表和表所示。

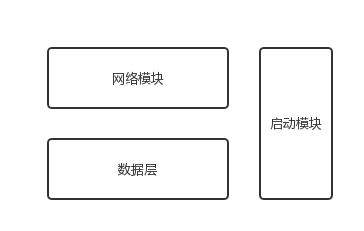
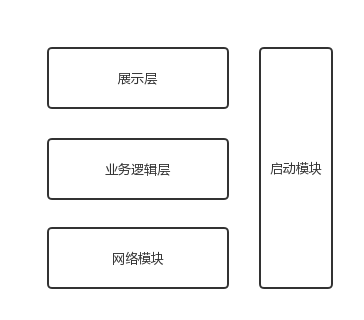


图 客户端模块视图 图 服务器端模块视图

**表 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 | 基于窗口的互联网酒店管理系统客户端用户界面 |
| 业务逻辑层 | 对于用户界面的输入进行响应并进行业务处理逻辑 |
| 客户端网络模块 | 利用Java RMI机制查找RMI服务 |

**表 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。层之间调用的接口如表所示。

**表 层之间调用的接口**

|  |  |  |
| --- | --- | --- |
| 接口 | 服务调用方 | 服务提供方 |
| OrderBLService  HotelBLService  MemberBLService  AccountBLService  PromotionBLService | 客户端展示层 | 客户端业务逻辑层 |
| OrderDataService  HotelDataService  MemberDataService  AccountDataService  PromotionDataService  DatabaseFactory | 客户端业务逻辑层 | 服务器端数据层 |

借用客户用例来说明层之间的调用，如图所示。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）。MemberBLService提供了Member界面所需要的所有业务逻辑功能。MemberDataService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。

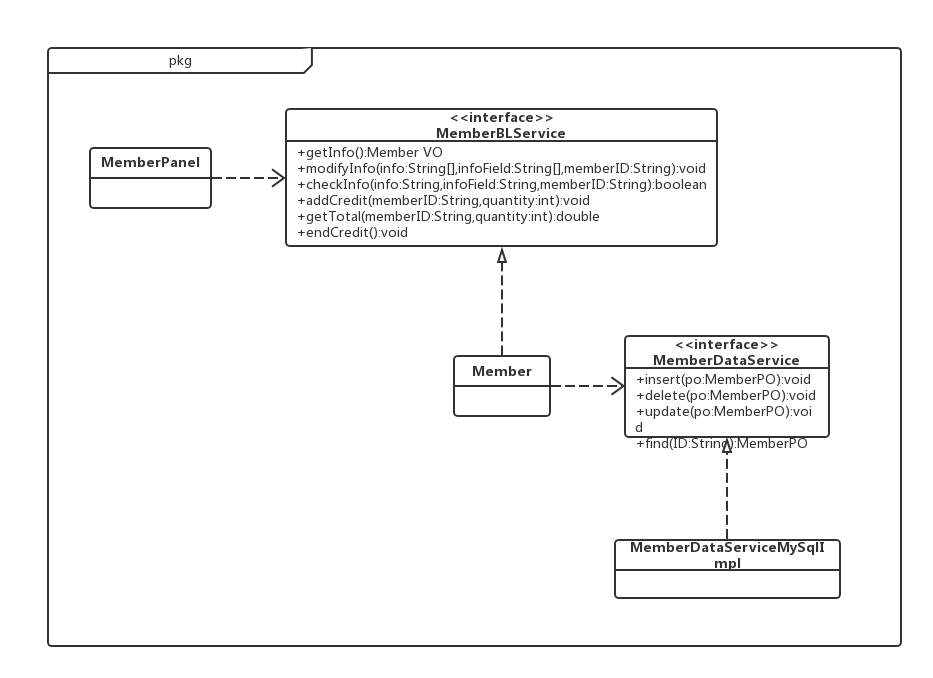


图 客户用例层之间调用的接口

* 1. 用户界面层的分解

根据需求，系统存在23个用户界面：登陆界面、客户主界面、酒店主界面、网站营销人员主界面、网站管理人员主界面、个人基本信息界面、评价酒店服务界面、订单信息界面、提交订单界面、撤销订单界面、注册会员界面、获取酒店信息界面、酒店基本信息、执行客户订单界面、恢复客户订单界面、更新退房信息界面、更新入住信息界面、制定酒店促销策略界面、制定网站促销策略界面、信用充值界面、撤销异常订单界面、分析未执行订单情况界面、调整用户界面。界面跳转如图所示。

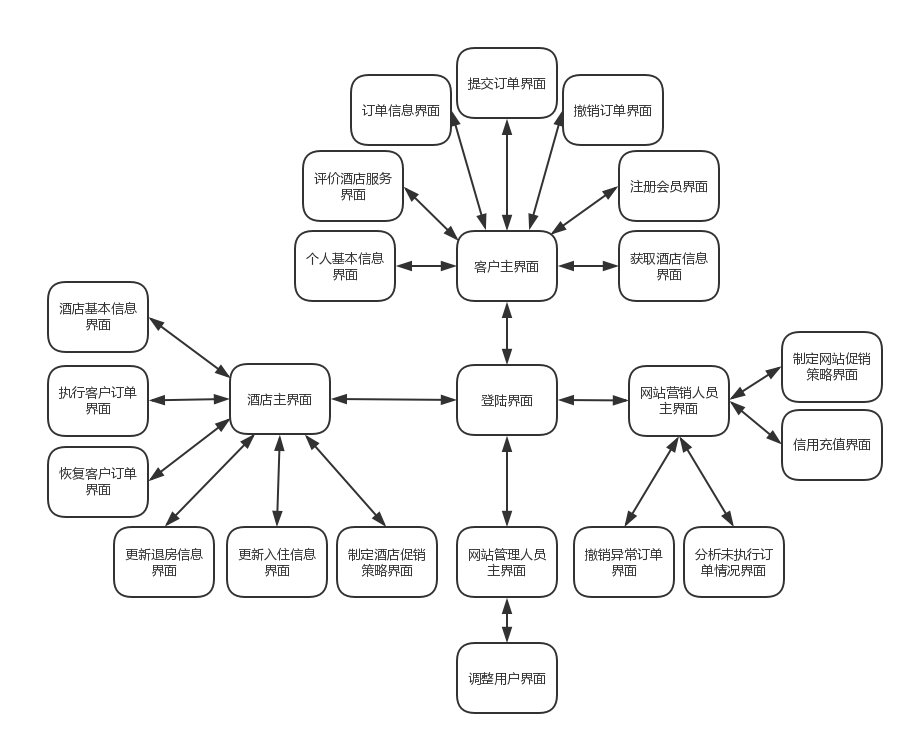


图 用户界面跳转

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。用户界面类如图所示。

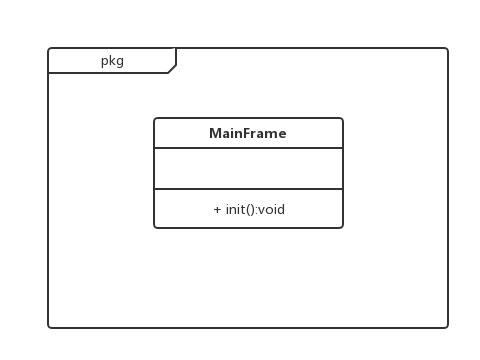


图 用户界面类

* + 1. 用户界面层模块的职责

如表所示为用户界面层模块的职责

**表 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 |

* + 1. 用户界面层模块的接口规范

用户界面层模块的接口规范如表所示。

**表 用户界面层模块的接口规范**

|  |  |  |
| --- | --- | --- |
| MainFrame | 语法 | Init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示Frame以及LoginPanel |

用户界面层需要的服务接口如表所示。

**表 用户界面层模块需要的服务接口**

|  |  |
| --- | --- |
| 服务名 | 服务 |
| businesslogicservice.LoginBLService | 登陆界面的业务逻辑接口 |
| Businessloginservice.\*BLService | 每个界面都有一个相应的业务逻辑接口 |

* + 1. 用户界面模块设计原理

用户界面利用JavaFX的API来实现

* 1. 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，Order对象负责处理订单界面的业务逻辑。业务逻辑层的设计如图所示。

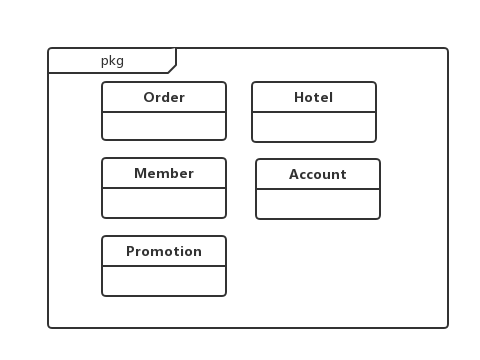


图 业务逻辑层的设计

* + 1. 业务逻辑层模块的职责

业务逻辑层模块职责如表所示

表 业务逻辑层模块的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| orderbl |  |
| hotelbl |  |
| memberbl |  |
| accountbl |  |
| promotionbl |  |

* + 1. 业务逻辑层模块的接口规范

orderbl、hotelbl、memberbl、accountbl和promotionbl模块的接口规范分别如表、表、表、表和表所示

**表 memberbl模块的接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| Member.modifyInfo | 语法 | Public ResultMessage modifyInfo(String[] info, String[] infoField, String memberID) | |
| 前置条件 | info符合输入规则 | |
| 后置条件 | 修改用户信息，持久化更新涉及的领域对象的数据 | |
| Member.checkInfo | 语法 | Public ResultMessage checkInfo(String Info, String InfoField, String memberID) | |
| 前置条件 | Info被修改或特定infoField为空 | |
| 后置条件 | 返回检查结果 | |
| Member.addCredit | 语法 | Public ResultMessage addCredit(String memberID, int quantity) | |
| 前置条件 | 启动一个信用充值回合 | |
| 后置条件 | 在一个信用充值回合中，增加信用充值的数量 | |
| Member.getTotal | 语法 | Public ResultMessage getTotal(String memberID, int quantity) | |
| 前置条件 | 已添加充值者信息、充值信用值数量 | |
| 后置条件 | 返回此信用充值回合中需要支付的总额 | |
| Member.endCredit | 语法 | Public void endCredit() | |
| 前置条件 | 已支付 | |
| 后置条件 | 结束此次信用充值回合，持久化更新涉及的领域对象的数据 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | **服务** |
| MemberDataService.find(String ID) | | | 根据ID进行查找单一持久化对象 |
| MemberDataService.insert(MemberPO po) | | | 插入单一持久化对象 |
| MemberDataService.delete(MemberPO po) | | | 删除单一持久化对象 |
| MemberDataService.update(MemberPO po) | | | 更新单一持久化对象 |
| DataBaseFactory.getMemberDatabase | | | 得到Member数据库的服务的引用 |

* 1. 数据层的分解

1. 信息视角
   1. 数据持久化对象
   2. Txt持久化格式
   3. 数据库表