目录

[更新历史 2](#_Toc465540260)

[一、 引言 4](#_Toc465540261)

[(一) 编制目的 4](#_Toc465540262)

[(二) 词汇表 4](#_Toc465540263)

[(三) 参考资料 5](#_Toc465540264)

[二、 产品概述 6](#_Toc465540265)

[三、 体系结构设计概述 7](#_Toc465540266)

[四、 结构视角 8](#_Toc465540267)

[(一) 展示层的分解 8](#_Toc465540268)

[1. 用户界面层模块的职责 9](#_Toc465540269)

[2. 用户界面模块的接口规范 9](#_Toc465540270)

[3. 用户界面层模块的接口规范 9](#_Toc465540271)

[4. 用户界面模块设计原理 10](#_Toc465540272)

[(二) 业务逻辑层的分解 10](#_Toc465540273)

[1. logIn模块 10](#_Toc465540274)

[2. order模块 14](#_Toc465540275)

[(三) 数据层的分解 21](#_Toc465540276)

[1. 模块 21](#_Toc465540277)

[2. 模块 21](#_Toc465540278)

[五、 依赖视角 21](#_Toc465540279)

# 更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| **修改人员** | **修改日期** | **修改原因** | **版本号** |
| 冯俊杰 | 2016/10/4 | 最初草稿 |  |
| 全体成员 | 2016/10/10 | 新增逻辑分包 |  |
| 全体成员 | 2016/10/11 | 新增物理分包 |  |
| 冯俊杰 | 2016/10/12 | 修改物理分包 |  |
| 龚尘淼 | 2016/10/12 | 修改逻辑分包 |  |
| 龚尘淼 | 2016/10/12 | 新增promotion、hotel接口规范 |  |
| 董金玉 | 2016/10/12 | 新增member、user接口规范 |  |
| 冯俊杰 | 2016/10/12 | 新增logIn、credit、market接口规范 |  |
| 高源 | 2016/10/12 | 新增order接口规范 |  |
| 董金玉 | 2016/10/13 | member、user接口规范评审后第一次更新 |  |
| 冯俊杰 | 2016/10/13 | logIn、credit、market接口规范评审后第一次更新 |  |
| 高源 | 2016/10/13 | order接口规范评审后第一次更新 |  |
| 冯俊杰 | 2016/10/13 | credit、market接口规范评审后第二次更新 |  |
| 冯俊杰 | 2016/10/13 | 修改物理分包逻辑层依赖关系 |  |
| 龚尘淼 | 2016/10/14 | 修改逻辑图和用户界面跳转图 |  |
| 龚尘淼 | 2016/10/14 | promotion、hotel接口规范评审后第一次更新 |  |
| 冯俊杰 | 2016/10/14 | 修改逻辑分包 |  |
| 高源 | 2016/10/14 | 新增进程图 |  |
| 龚尘淼 | 2016/10/14 | promotion、hotel接口规范评审后第二次更新 |  |
| 董金玉 | 2016/10/14 | member、user接口规范评审后第二次更新 |  |
| 高源 | 2016/10/15 | order接口规范评审后第二次更新 |  |
| 冯俊杰 | 2016/10/15 | 整合、合并文档 | V0.1 |
| 龚尘淼 | 2016/10/14 | promotion、hotel接口规范评审后第三次更新 | V0.2 |
| 高源 | 2016/10/15 | 新增数据持久化对象PO | V0.3 |
| 高源 | 2016/10/16 | 修改更新数据持久化对象PO | V0.3.1 |
| 全体人员 | 2016/10/16 | 修改细节 | V0.4 |
| 高源 | 2016/10/16 | 排版 | V1.0 |
| 全体人员 | 2016/10/25 | 修改VO、PO | V1.1 |

# 引言

## 编制目的

本报告详细完成对互联网酒店预订系统的详细设计，达到指导后续软件构造的目的，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。

## 词汇表

|  |  |
| --- | --- |
| **缩写或单词** | **解释** |
| HRS | 互联网酒店预订系统 |
| order | 订单 |
| hotel | 酒店 |
| guest | 客户 |
| hotelWorker | 酒店工作人员 |
| webMarketer | 网站营销人员 |
| webManager | 网站管理人员 |
| signUp | 注册 |
| logIn | 登录 |
| UI | 界面 |
| BL | 逻辑 |
| Data | 数据 |
| promotion | 促销策略 |
| credit | 信用值 |
| member | 会员 |
| market | 营销 |
|  |  |
|  |  |
|  |  |
|  |  |

## 参考资料

1. 互联网酒店预订系统用例文档
2. 互联网酒店预订系统需求规格说明文档
3. 互联网酒店预订系统体系结构设计文档
4. 丁二玉，刘钦.计算与软件工程（卷二）[M]机械工业出版2012：134—182
5. IEEE std 1471-2000

# 产品概述

参考《互联网酒店预订系统用例文档》和《互联网酒店预订系统需求规格说明文档》中对产品的概括描述。互联网酒店预订系统主要功能见用例图如下。



# 体系结构设计概述

参考互联网酒店预订系统体系结构设计文档的描述。

主要采用分层的架构方式，采用分布式的部署方式，将数据存储在服务器端，客户端。

通过RMI方式调用获取数据。

# 结构视角

## 展示层的分解

根据需求，系统存在38个用户界面（加粗为主界面）： **登录界面**、**注册界面**、**客户主界面**、个人基本信息界面、个人基本信息维护界面、会员注册界面、未评价订单界面、订单评价界面、个人订单概况列表界面、个人订单详情界面、个人订单撤销界面、商圈地址选定界面、酒店概况列表界面、订单生成界面、酒店搜索界面、酒店详情界面、会员注册界面、**酒店工作人员界面**、酒店信息界面、酒店信息维护界面、客房信息维护界面、酒店促销策略界面、订单列表界面、订单详情界面、剩余客房信息界面、线下客户入住和退房处理界面、**网站营销人员界面**、用户编号录入界面、信用充值界面、异常订单概况列表界面、异常订单详情界面、异常订单撤销界面、网站促销策略界面、会员等级制定界面、**网站管理人员界面**、会员信息界面、会员信息维护界面、酒店信息添加界面。

界面跳转如下图所示。



### 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 |

### 用户界面模块的接口规范

### 用户界面层模块的接口规范

|  |  |  |
| --- | --- | --- |
| mainUI | 语法 | init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示Frame |
| logInUI | 语法 | init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示LoginPanel |
| guestUI | 语法 | init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示guestPanel |
| hotelWorkerUI | 语法 | init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示hotelWorkerPanel |
| webMarketerUI | 语法 | init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示webMarketerPanel |
| webManagerUI | 语法 | init(args:String[]) |
| 前置条件 | 无 |
| 后置条件 | 显示webManagerPanel |

1. **用户界面层模块需要的服务接口**

|  |  |
| --- | --- |
| 服务名 | 服务 |
| businessLogicService.logInBLService | 登录界面和客户注册的业务逻辑接口 |
| businessLogicService.orderBLService | 负责处理与订单相关的接口 |
| businessLogicService.hotelBLService | 负责处理与酒店相关的接口 |
| businessLogicService.promotionBLService | 负责处理与促销策略相关的接口 |
| businessLogicService.creditBLService | 负责处理与信用值相关的接口 |
| businessLogicService.memberBLService | 负责处理与会员相关的接口 |
| businessLogicService.userBLService | 负责处理与所有用户信息相关的接口 |
| businessLogicService.marketBLService | 负责处理与会员等级制度信息相关的接口 |

### 用户界面模块设计原理

用户界面拟用java的Swing和AWT库来实现。

## 业务逻辑层的分解

业务逻辑层的开发包图请查看体系结构设计文档。

### logIn模块

1. 模块概述

logIn模块承担的需求参见需求规格说明文档功能需求及相关非功能需求。

logIn模块的职责和接口参见软件系统结构描述文档内对该模块的描述。

1. 整体结构

根据体系结构的设计，我们将系统分为展示层、业务逻辑层、数据层。每一层之间为了增加灵活性，我们会添加接口。在展示层和业务逻辑层之间添加logInBLService接口。为了隔离业务逻辑职责和逻辑控制职责，添加logInController，这样logInController会将对用户登录和客户注册的业务逻辑处理委托给logIn对象。guestPO、hotelWorkerPO、webMarketer、webManager是作为账户信息的持久性对象被添加到设计模型中去的。

1. 模块内部类的接口规范
2. Controller类的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| logInController.guestLogIn | 语法 | Public ResultMessage guestLogIn (String guest, String password) |
| 前置条件 | 客户输入客户编号和密码 |
| 后置条件 | 客户被认证并进入系统客户主界面 |
| logInController.hotelWorkerLogIn | 语法 | Public ResultMessage hotelWorkerLogIn (String hotelWorker, String password) |
| 前置条件 | 酒店工作人员输入编号和密码 |
| 后置条件 | 酒店工作人员被认证并进入系统酒店工作人员主界面 |
| logInController.webMarketerLogIn | 语法 | Public ResultMessage webMarketerLogIn (String webMarketer, String password) |
| 前置条件 | 网站营销人员输入编号和密码 |
| 后置条件 | 网站营销人员被认证并进入系统网站营销人员主界面 |
| logInController.webManagerLogIn | 语法 | Public ResultMessage webManagerLogIn (String webManager, String password) |
| 前置条件 | 网站管理人员输入编号和密码 |
| 后置条件 | 网站管理人员被认证并进入系统网站管理人员主界面 |
| logInController.guestSignUp | 语法 | Public ResultMessage guestSignUp (GuestVO guestVO) |
| 前置条件 | 客户信息符合规范要求 |
| 后置条件 | 系统返回是否成功添加客户；是则进入系统的客户主界面，否则提示添加失败 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| logIn.guestLogIn | 客户登录系统 | |
| logIn.hotelWorkerLogIn | 酒店工作人员登录系统 | |
| logIn.webMarketerLogIn | 网站营销人员登录系统 | |
| logIn.webManagerLogIn | 网站管理人员登录系统 | |
| logIn.guestSignUp | 客户注册 | |

1. Domain类的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| logIn.guestLogIn | 语法 | Public ResultMessage guestLogIn (String guest, String password) |
| 前置条件 | 客户输入客户编号和密码 |
| 后置条件 | 客户被认证并进入系统客户主界面 |
| logIn.hotelWorkerLogIn | 语法 | Public ResultMessage hotelWorkerLogIn (String hotelWorker, String password) |
| 前置条件 | 酒店工作人员输入编号和密码 |
| 后置条件 | 酒店工作人员被认证并进入系统酒店工作人员主界面 |
| logIn.webMarketerLogIn | 语法 | Public ResultMessage webMarketerLogIn (String webMarketer, String password) |
| 前置条件 | 网站营销人员输入编号和密码 |
| 后置条件 | 网站营销人员被认证并进入系统网站营销人员主界面 |
| logIn.webManagerLogIn | 语法 | Public ResultMessage webManagerLogIn (String webManager, String password) |
| 前置条件 | 网站管理人员输入编号和密码 |
| 后置条件 | 网站管理人员被认证并进入系统网站管理人员主界面 |
| logIn.guestSignUp | 语法 | Public ResultMessage guestSignUp (GuestVO guestVO) |
| 前置条件 | 客户信息符合规范要求 |
| 后置条件 | 系统返回是否成功添加客户；是则进入系统的客户主界面，否则提示添加失败 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| User.getSingle(String userID,UserType userType) | 返回与该ID对应的user信息 | |
| User.add(UserPO newUserPO) | 新增该用户并持久化保存该用户的数据 | |
| User.getLogInfo(String userID, UserType userType) | 返回与该ID对应的user密码 | |

1. 业务逻辑层的动态模型
2. 业务逻辑层的设计原理

### order模块

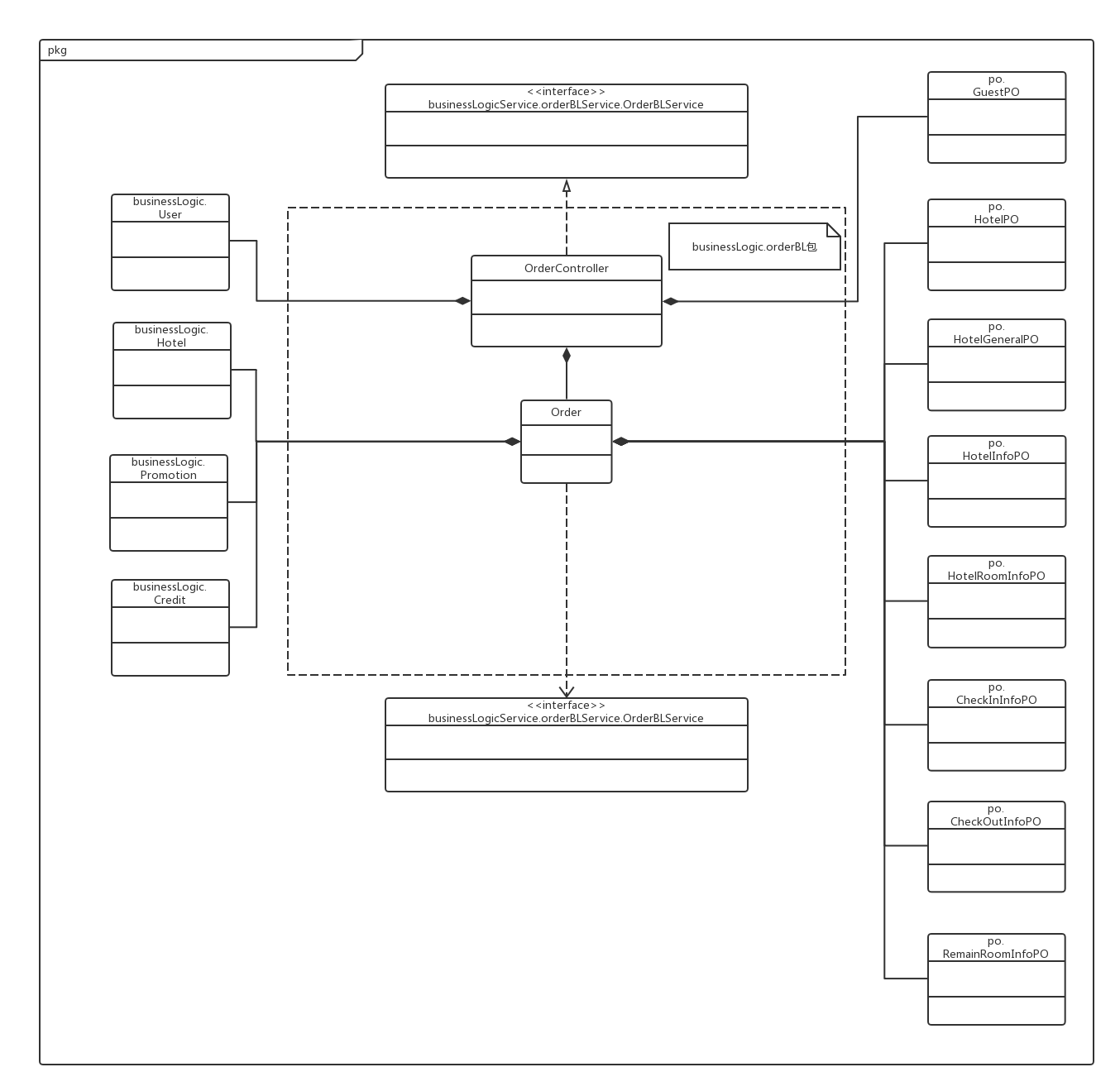
1. 模块概述

order模块承担的需求参见需求规格说明文档功能需求及相关非功能需求。

order模块的职责和接口参见软件系统结构描述文档内对该模块的描述。

1. 整体结构

根据体系结构的设计，我们将系统分为展示层、业务逻辑层、数据层。每一层之间为了增加灵活性，我们会添加接口。在展示层和业务逻辑层之间添加orderBLService接口，在业务逻辑层和数据层之间添加orderDataService接口。为了隔离业务逻辑职责和逻辑控制职责，添加OrderController，这样OrderController会将对用户登录和客户注册的业务逻辑处理委托给Order对象。GuestPO、HotelPO、HotelGeneralPO、HotelInfoPO、HotelRoomInfoPO、CheckInInfoPO、CheckOutInfoPO、RemainRoomInfoPO是作为账户信息的持久性对象被添加到设计模型中去的。



1. 模块内部类的接口规范
2. Controller类的接口规范

|  |  |  |
| --- | --- | --- |
| 1. 提供的服务（供接口） | | |
| OrderController.createOrder | 语法 | Public ResultMessage createOrder(orderVO OrderVO) |
| 前置条件 | 客户填写订单的内容符合规则且完整 |
| 后置条件 | 系统生成未执行订单并持久化增加该客户的订单数据 |
| OrderController.executeOrder | 语法 | Public ResultMessage executeOrder (String orderID) |
| 前置条件 | 订单状态为未执行 |
| 后置条件 | 系统更改状态为已执行订单并持久化保存该客户的订单数据 |
| OrderController.undoAbnormalOrder | 语法 | Public ResultMessage undoAbnormalOrder(String orderID) |
| 前置条件 | 订单状态为异常 |
| 后置条件 | 系统更改状态为已撤销订单并持久化保存该客户的订单数据 |
| OrderController.undoNormalOrder | 语法 | Public ResultMessage undoNormalOrder(String orderID) |
| 前置条件 | 订单状态为未执行 |
| 后置条件 | 系统更改状态为已执行订单并持久化保存该客户的订单数据 |
| OrderController.getOrderDetail | 语法 | Public OrderVO getOrderDetail(String orderID) |
| 前置条件 | 知道订单号，需要知道订单内容 |
| 后置条件 | 系统返回该订单号的订单 |
| OrderController.getAllGuestOrderGeneral | 语法 | Public List<OrderGeneralVO> getAllGuestOrderGeneral (String guestID) |
| 前置条件 | 已经登录，需要知道的订单号 |
| 后置条件 | 系统返回查找到的订单结果 |
| OrderController.getAllHotelOrderGeneral | 语法 | Public List<OrderGeneralVO> getAllHotelOrderGeneral (String hotelID) |
| 前置条件 | 酒店工作人员已经登录 |
| 后置条件 | 系统返回该酒店所有订单 |
| OrderController.getBookedHotel | 语法 | Public List<String> getBookedHotel (String guestID) |
| 前置条件 | 客户需要知道所有住过酒店的信息 |
| 后置条件 | 系统返回所有该客户已住过酒店的ID列表 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| Order.createOrder(orderVO OrderVO) | 订单生成 | |
| Order.executeOrder (String orderID) | 订单执行 | |
| Order.undoAbnormalOrder (String orderID) | 撤销异常订单 | |
| Order.undoNormalOrder | 撤销正常订单 | |
| Order.getOrderDetail (String orderID) | 查看订单概况 | |
| Order.getAllGuestOrderGeneral(String guestID) | 查看所有客户订单概况 | |
| Order.getAllHotelOrderGeneral(String hotelID) | 查看所有酒店订单概况 | |
| Order.getBookedHotel(String guestID) | 查看客户预定过的酒店 | |
| User.getSingle(String userID,UserType userType) | 查看客户详情 | |

1. Domain类的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| Order.createOrder | 语法 | Public ResultMessage createOrder(orderPO OrderPO) |
| 前置条件 | 客户填写订单的内容符合规则且完整 |
| 后置条件 | 系统生成未执行订单并持久化增加该客户的订单数据 |
| Order.executeOrder | 语法 | Public ResultMessage executeOrder (String orderID) |
| 前置条件 | 订单状态为未执行 |
| 后置条件 | 系统更改状态为已执行订单并持久化保存该客户的订单数据 |
| Order.undoAbnormalOrder | 语法 | Public ResultMessage undoAbnormalOrder(String orderID) |
| 前置条件 | 订单状态为异常 |
| 后置条件 | 系统更改状态为已撤销订单并持久化保存该客户的订单数据 |
| Order.undoNormalOrder | 语法 | Public ResultMessage undoNormalOrder(String orderID) |
| 前置条件 | 订单状态为未执行 |
| 后置条件 | 系统更改状态为已执行订单并持久化保存该客户的订单数据 |
| Order.getOrderDetail | 语法 | Public OrderVO getOrderDetail(String orderID) |
| 前置条件 | 知道订单号，需要知道订单内容 |
| 后置条件 | 系统返回该订单号的订单 |
| Order.getAllGuestOrderGeneral | 语法 | Public List<OrderGeneralVO> getAllGuestOrderGeneral (String guestID) |
| 前置条件 | 已经登录，需要知道的订单号 |
| 后置条件 | 系统返回查找到的订单结果 |
| Order.getAllHotelOrderGeneral | 语法 | Public List<OrderGeneralVO> getAllHotelOrderGeneral (String hotelID) |
| 前置条件 | 酒店工作人员已经登录 |
| 后置条件 | 系统返回该酒店所有订单 |
| Order.getBookedHotel | 语法 | Public List<String> getBookedHotel (String guestID) |
| 前置条件 | 客户需要知道所有住过酒店的信息 |
| 后置条件 | 系统返回所有该客户已住过酒店的ID列表 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| OrderDataService.createOrder(orderVO OrderVO) | 根据提供的po增加持久化对象 | |
| OrderDataService.executeOrder(String orderID) | 修改单一持久化对象 | |
| OrderDataService.getOrderDetail(String orderID) | 根据ID查找持久化对象 | |
| OrderDataService.getAllHotelOrderGeneral(String hotelID) | 根据ID查找持久化对象 | |
| OrderDataService.getAllGuestOrderGeneral(String guestID) | 根据ID查找持久化对象 | |
| OrderDataService.undoNormalOrder(String orderID) | 修改单一持久化对象 | |
| OrderDataService.UndoAbnormalOrder(String orderID) | 修改单一持久化对象 | |
| Hotel.getHotelInfo(String userID) | 获得单一对象 | |
| Promotion.getDiscount(PreOrder preOrder) | 获得单一对象 | |

1. 业务逻辑层的动态模型
2. 业务逻辑层的设计原理

## 数据层的分解

数据层模块的分解详见需求规格文档中的相关描述。

### 模块

### 模块

# 依赖视角