### hotel模块

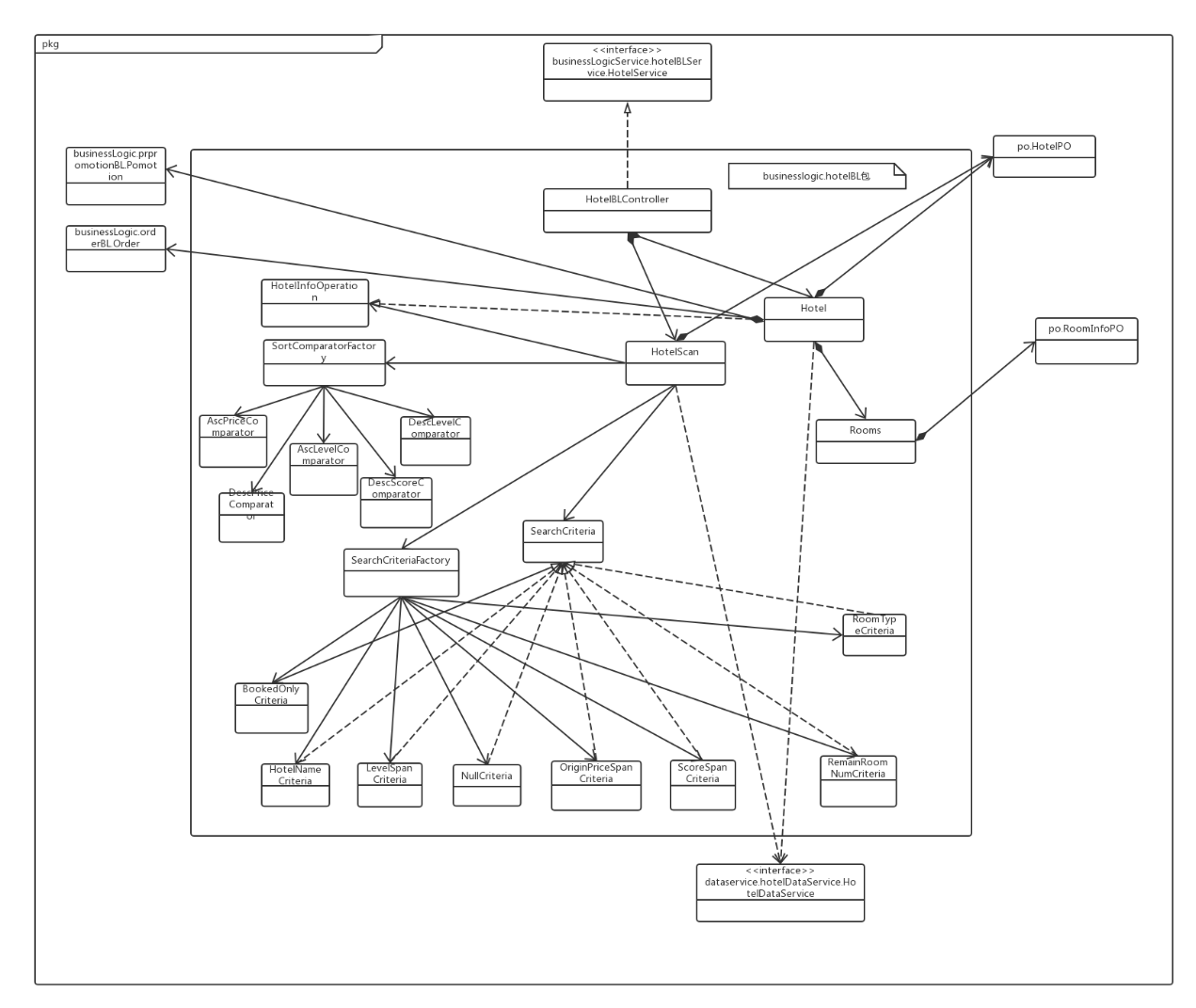
1. 模块概述

hotel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。

hotel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hotelBLService接口。为了隔离业务逻辑职责和逻辑控制职责，添加hotelBLService，这样hotelBLService会依赖于ExecutedOrderInfo、UnexecutedOrderInfo、AbnormalOrderInfo三个接口实现对订单执行以及退房的业务逻辑，同时hotelBLService将对酒店基本信息维护、客房信息维护、剩余客房信息查看、酒店促销策略维护、线下处理的业务逻辑委托给hotel对象。HotelPO、HotelWorkerPO、RoomInfo、RemainRoomInfo是作为酒店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

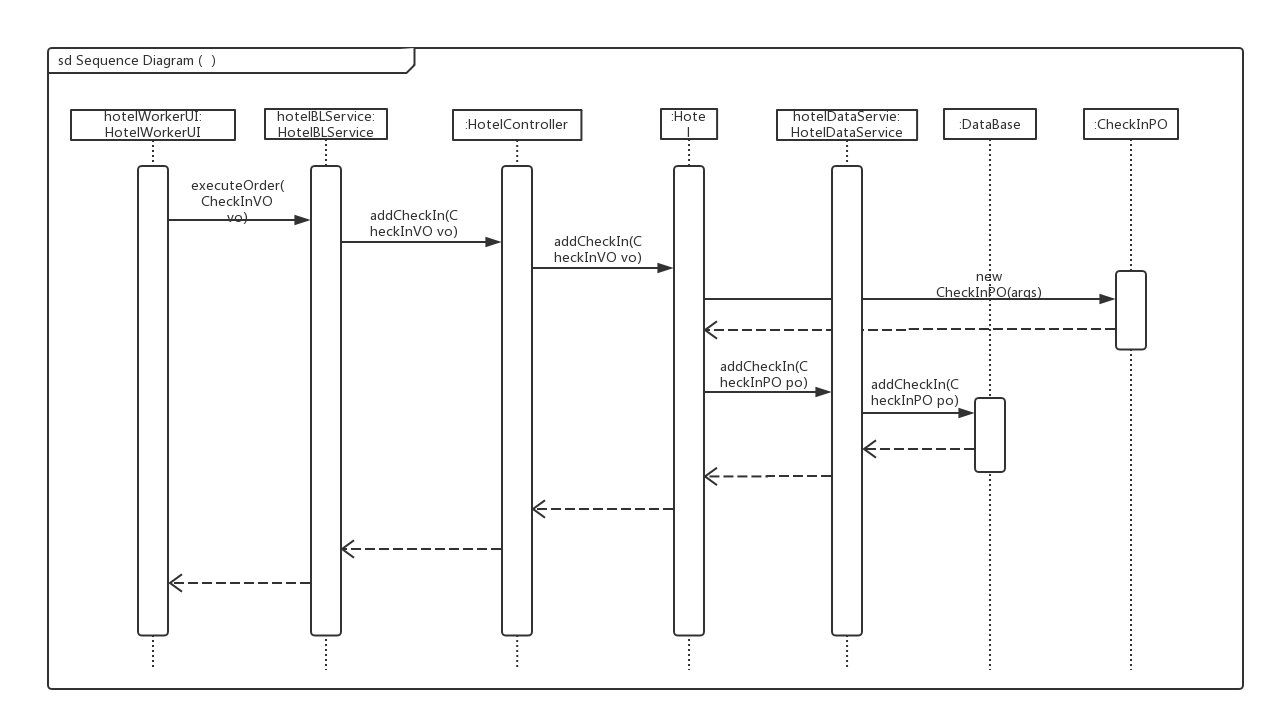
|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| HotelController.getHotelInfo | 语法 | public HotelVO getHotelInfo(String hotelID) |
| 前置条件 | 客户或酒店工作人员已被识别和授权 |
| 后置条件 | 无 |
| HotelController.updateHotelInfo | 语法 | public ResultMessage updateHotelInfo (HotelVO hotelVO) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 系统更新并持久化保存酒店基本信息 |
| HotelController.addHotel | 语法 | public ResultMessage addHotel(HotelVO hotelVO) |
| 前置条件 | 网站管理人员已被识别和授权 |
| 后置条件 | 系统添加并持久化保存一条酒店基本信息 |
| HotelController.addRoomType | 语法 | public ResultMessage addRoomType(RoomInfoVO roomInfoVO) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 系统添加并持久化保存一条客房信息 |
| HotelController.getHotelRoomInfo | 语法 | public Iterator<RoomInfoVO> getHotelRoomInfo (String hotelWorkerID) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 无 |
| HotelController.updateHotelRoomInfo | 语法 | public ResultMessage updateHotelRoomInfo (RoomInfoVO roomInfoVO) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 系统更新并持久化保存一条客房信息 |
| HotelController.getRemainRoomNum | 语法 | public int getRemainRoomNum(String hotelID,RoomType roomType) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 无 |
| HotelController.getOriginPrice | 语法 | public int getOriginPrice(String hotelID, RoomType roomType) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 无 |
| HotelController.checkInOffline | 语法 | public ResultMessage checkInOffline(String hotelID,RoomType roomType, int roomNum) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 更新该酒店该房型的剩余房间数量 |
| HotelController.checkOutOffline | 语法 | public ResultMessage checkOutOffline(String hotelID,RoomType roomType, int roomNum) |
| 前置条件 | 酒店工作人员已被识别和授权 |
| 后置条件 | 更新该酒店该房型的剩余房间数量 |
| HotelController.getHotels | 语法 | public Iterator<HotelVO> getHotels(String city,String circle) |
| 前置条件 | 客户已被识别和授权 |
| 后置条件 |  |
| HotelController.sortHotels | 语法 | public Iterator<HotelVO> sortHotels(SortStrategy sortStrategy) |
| 前置条件 | 已知排序策略 |
| 后置条件 | 返回排好序的酒店集合 |
| HotelController.getSortedHotels | 语法 | public List<HotelGeneralVO> getSortedHotels (SortStrategy sortStrategy); |
| 前置条件 | 客户已被识别和授权 |
| 后置条件 | 无 |
| HotelController.searchHotels | 语法 | public Iterator<HotelVO> searchHotels(List<SearchCriteriaType> searchCriteriaTypes,SearchCriteriaVO vo) |
| 前置条件 | 客户已被识别和授权，筛选酒店 |
| 后置条件 | 返回经筛选的酒店集合 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| Hotel.getHotelInfo | 得到酒店基本信息 | |
| Hotel.updateHotelInfo | 更新酒店基本信息 | |
| Hotel.addHotelInfo | 添加酒店基本信息 | |
| Hotel.updateHotelRoomInfo | 更新酒店客房信息 | |
| Hotel.addRoomInfo | 添加酒店客房信息 | |
| Hotel.getHotelRoomInfo | 得到酒店客房信息集合 | |
| Hotel.getRemainNumOfSpecificType | 获取指定房型的剩余房间数量 | |
| Hotel.getOriginPrice | 获取指定房型的原始价格 | |
| Hotel.checkIn | 线下入住，更新客房剩余房间数量 | |
| Hotel.checkOut | 线下退房，更新客房剩余房间数量 | |
| HotelScan.getHotels | 获得指定城市、商圈的所有酒店 | |
| HotelScan.sortHotels | 对当前酒店列表排序 | |
| HotelScan.searchHotels | 对当前酒店列表进行筛选 | |

1. Domain类的接口规范

|  |
| --- |
| 提供的服务（供接口） |

|  |  |  |
| --- | --- | --- |
| hotel.getHotelInfo | 语法 | Public HotelVO getHotelInfo (String hotelID); |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 返回编号对应酒店的基本信息 |
| hotel.updateHotelInfo | 语法 | Public ResultMessage updateHotelInfo (HotelVO hotelVO) |
| 前置条件 | 已知需要更新的酒店基本信息 |
| 后置条件 | 系统更新该酒店基本信息，返回是否更新成功 |
| hotel.addHotelInfo | 语法 | Public ResultMessage addHotel (HotelVO hotelVO) |
| 前置条件 | 已知需要添加的酒店基本信息 |
| 后置条件 | 系统添加一条酒店基本信息，返回是否添加成功 |
| hotel.getHotelRoomInfo | 语法 | public Iterator<RoomInfoVO> getHotelRoomInfo(String hotelWorkerID) |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 返回该酒店的客房信息集合 |
| hotel.addRoomInfo | 语法 | public ResultMessage addRoomInfo(RoomInfoVO roomInfoVO) |
| 前置条件 | 已知一条客房信息 |
| 后置条件 | 系统添加一条客房信息，返回是否添加成功 |
| hotel.updateHotelRoomInfo | 语法 | public ResultMessage updateHotelRoomInfo(RoomInfoVO roomInfoVO) |
| 前置条件 | 已知一条客房信息 |
| 后置条件 | 系统更新该条客房信息，返回更新是否成功 |
| hotel.getLowestPrice | 语法 | public double getLowestPrice(String hotelID) |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 返回该酒店的最低房价 |
| hotel.getRemainNumOfSpecificType | 语法 | public int getRemainNumOfSpecificType(String hotelID, RoomType roomType) |
| 前置条件 | 已知酒店编号、房间类型 |
| 后置条件 | 返回该酒店该房型的剩余房间信息 |
| hotel.scoreUpdate | 语法 | public ResultMessage scoreUpdate(String hotelID,double score) |
| 前置条件 | 已知酒店编号、评分 |
| 后置条件 | 更新酒店评分 |
| hotel.getHotelAddress | 语法 | public Address getHotelAddress(String hotelID) |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 返回酒店的地址信息 |
| hotel.checkIn | 语法 | public ResultMessage checkIn(String hotelID, RoomType roomType, int roomNum) |
| 前置条件 | 线上入住，已知酒店编号、房间类型、房间数量 |
| 后置条件 | 更新该酒店该房型的剩余房间数量 |
| hotel.checkOut | 语法 | public ResultMessage checkOut(String hotelID, RoomType roomType, int roomNum) |
| 前置条件 | 线上退房，已知酒店编号、房间类型、房间数量 |
| 后置条件 | 更新该酒店该房型的剩余房间数量 |
| hotel.updateRemainRoomNumForUndoOrder | 语法 | public ResultMessage updateRemainRoomNumForUndoOrder(String hotelID, RoomType roomType, int roomNum) |
| 前置条件 | 客户撤销订单，已知酒店编号、房间类型、房间数量 |
| 后置条件 | 更新该酒店该房型的剩余房间数量 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| hotelDataService.getHotelInfo (String hotelID) | 返回对应该酒店编号的酒店基本信息 | |
| hotelDataService.updateHotelInfo(HotelPO hotelPO) | 更新数据库里的一条酒店基本信息 | |
| hotelDataService.addHotelInfo(HotelPO hotelPO) | 向数据库里添加一条酒店基本信息 | |
| Rooms.getRoomInfo(String hotelID) | 获取客房信息列表 | |
| Rooms.addRoomInfo(RoomInfoVO roomInfoVO) | 添加一条客房信息 | |
| Rooms.updateHotelRoomInfo(RoomInfoVO roomInfoVO) | 更新一条客房信息 | |
| Rooms.getLowestPrice(String hotelID) | 获取酒店最低价格 | |
| Rooms.getRemainNumOfSpecificType | 获取指定房型的剩余房间数量 | |
| Rooms.getRoomType(String hotelID) | 返回指定酒店的所有房型 | |
| Rooms.getRemainRoomNum(String hotelID) | 返回指定酒店剩余房间总数 | |
| Rooms.getRoomType(String hotelID) | 获取指定酒店的所有房型 | |
| Rooms.checkIn(String hotelID, RoomType roomType, int roomNum) | 入住，更新入住房型的剩余房间数量 | |
| Rooms.checkOut(String hotelID, RoomType roomType, int roomNum) | 退房，更新退房房型的剩余房间数量 | |
| Rooms.updateRemainRoomNumForUndoOrder(String hotelID, RoomType roomType, int roomNum) | 客户撤销未执行订单，更新该房型的剩余房间数量 | |

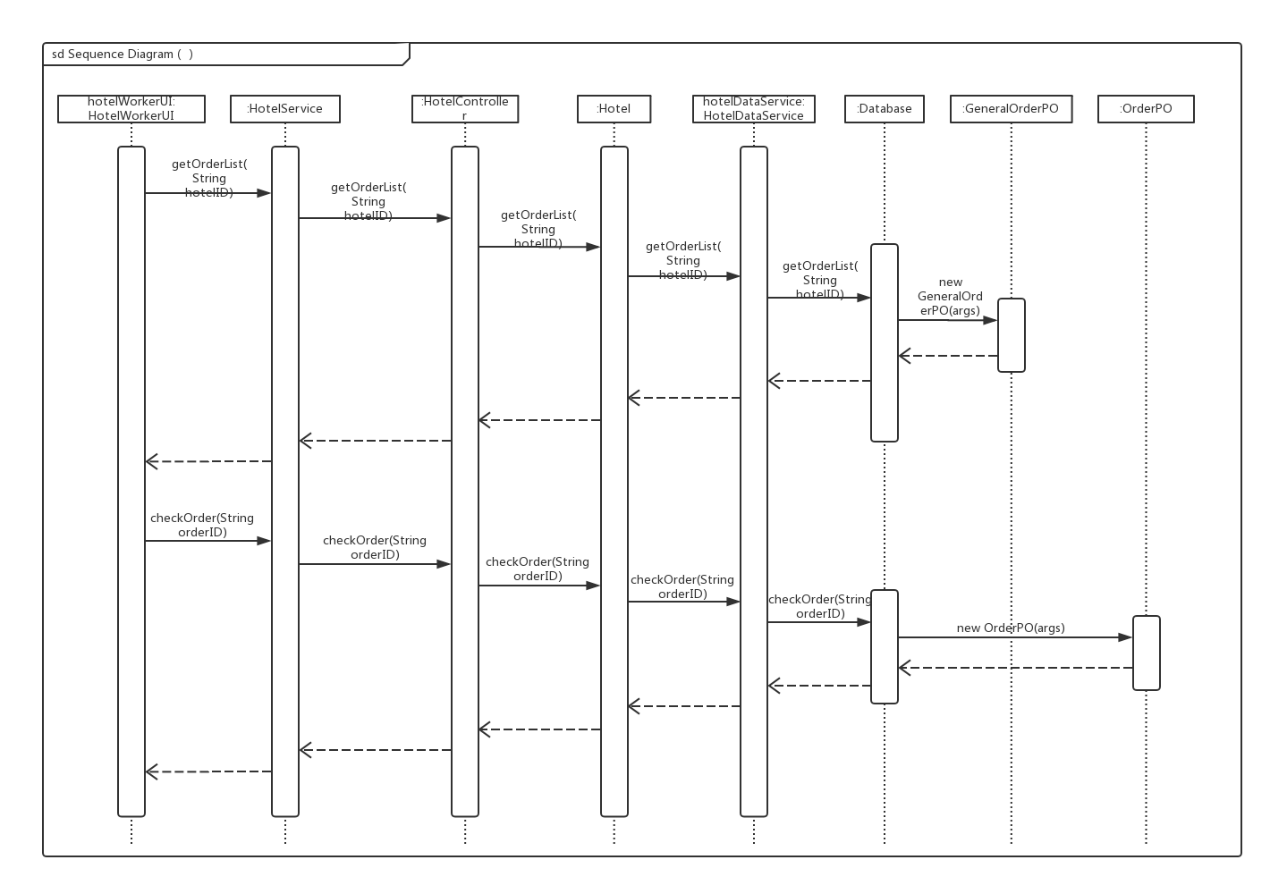
1. 业务逻辑层的动态模型



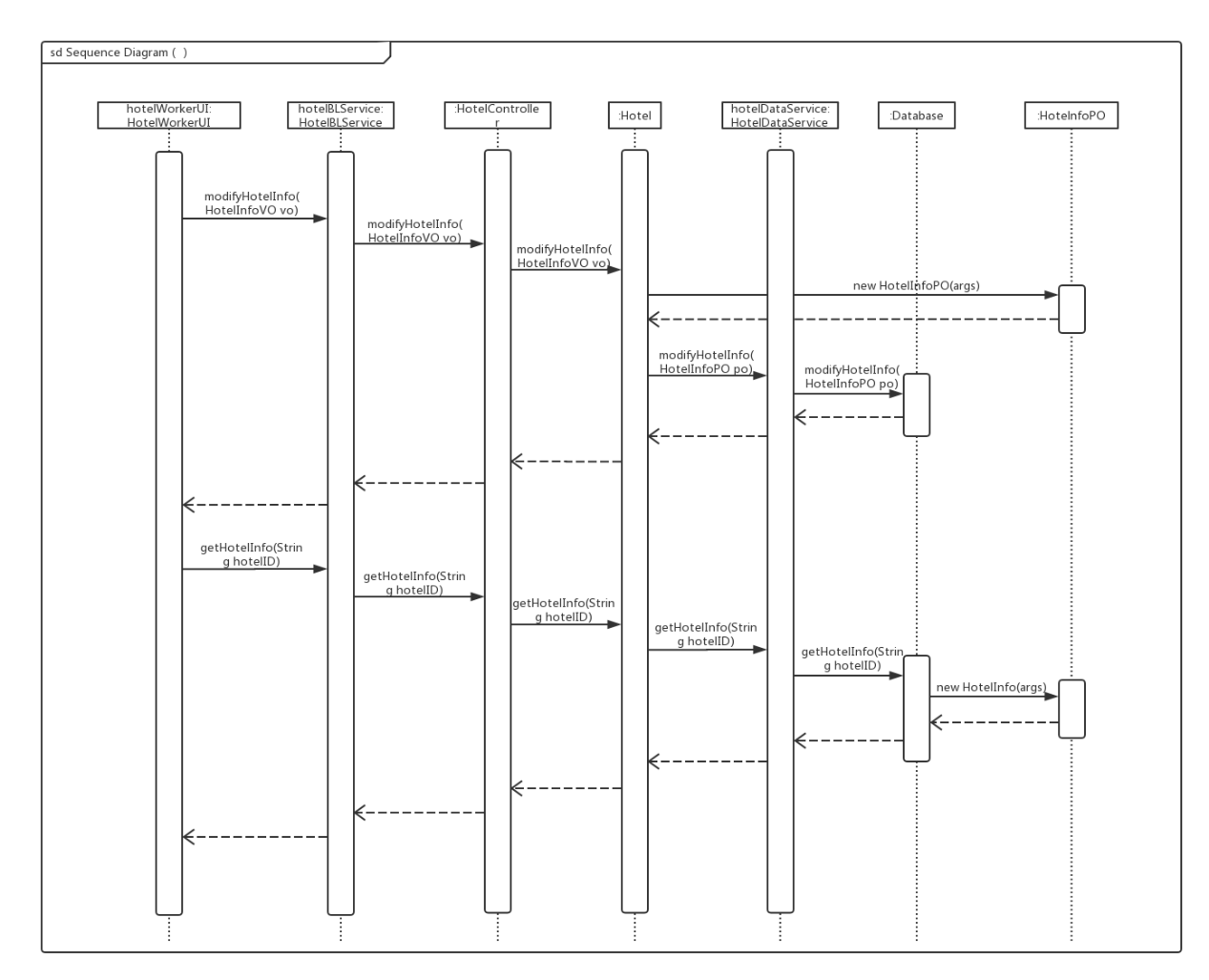
订单执行及更新入住信息顺序图



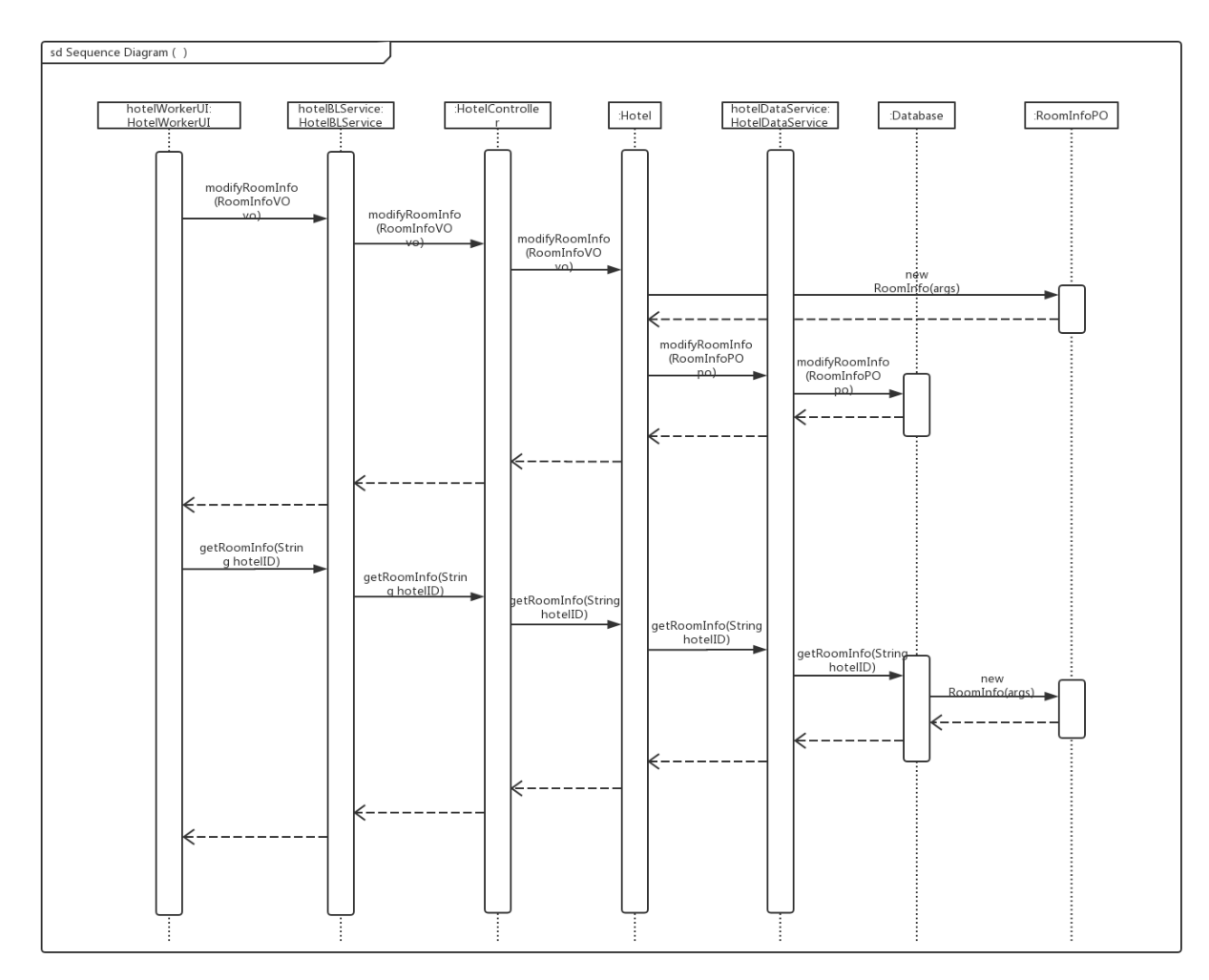
酒店促销策略维护顺序图



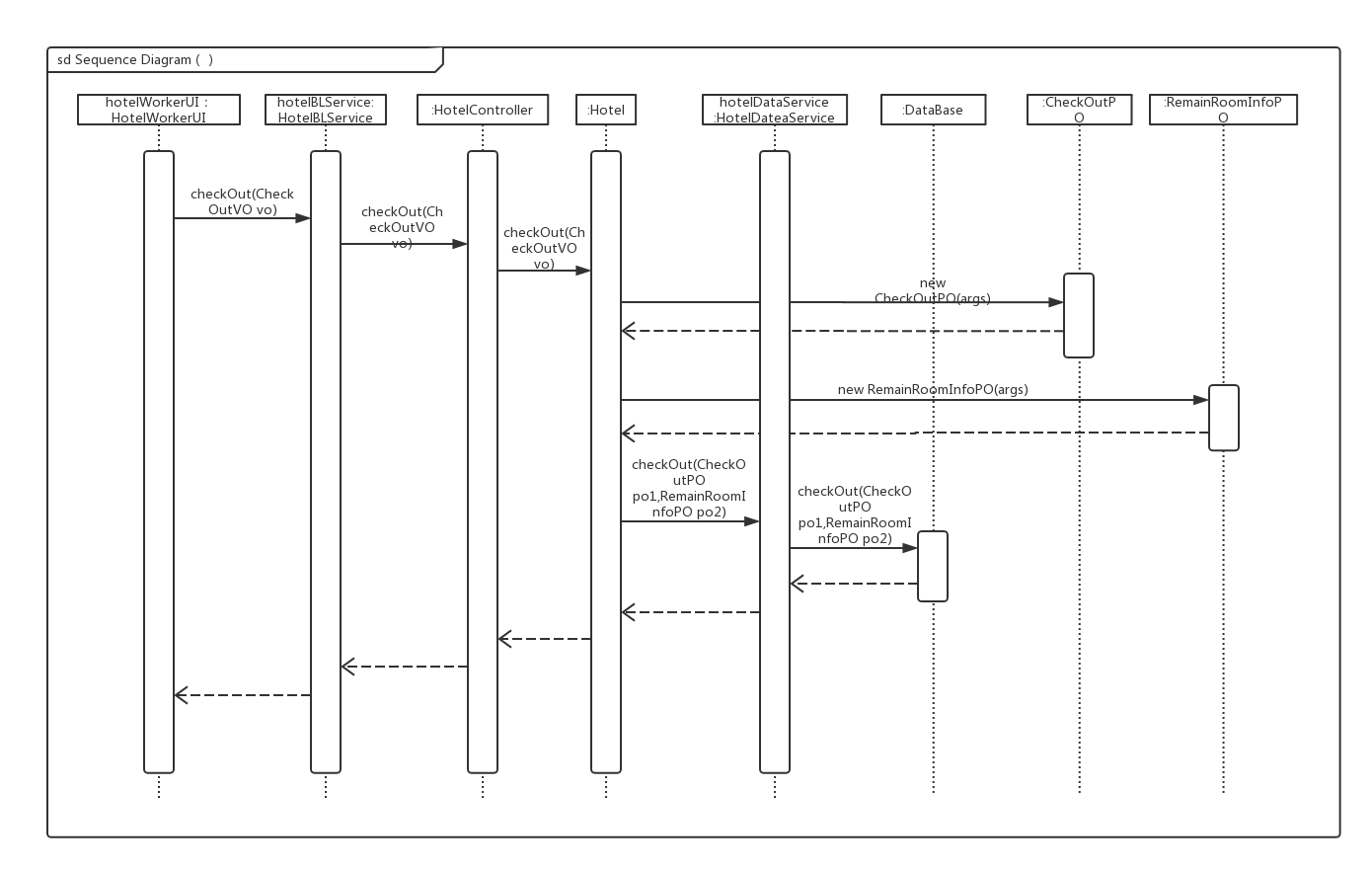
酒店订单浏览顺序图



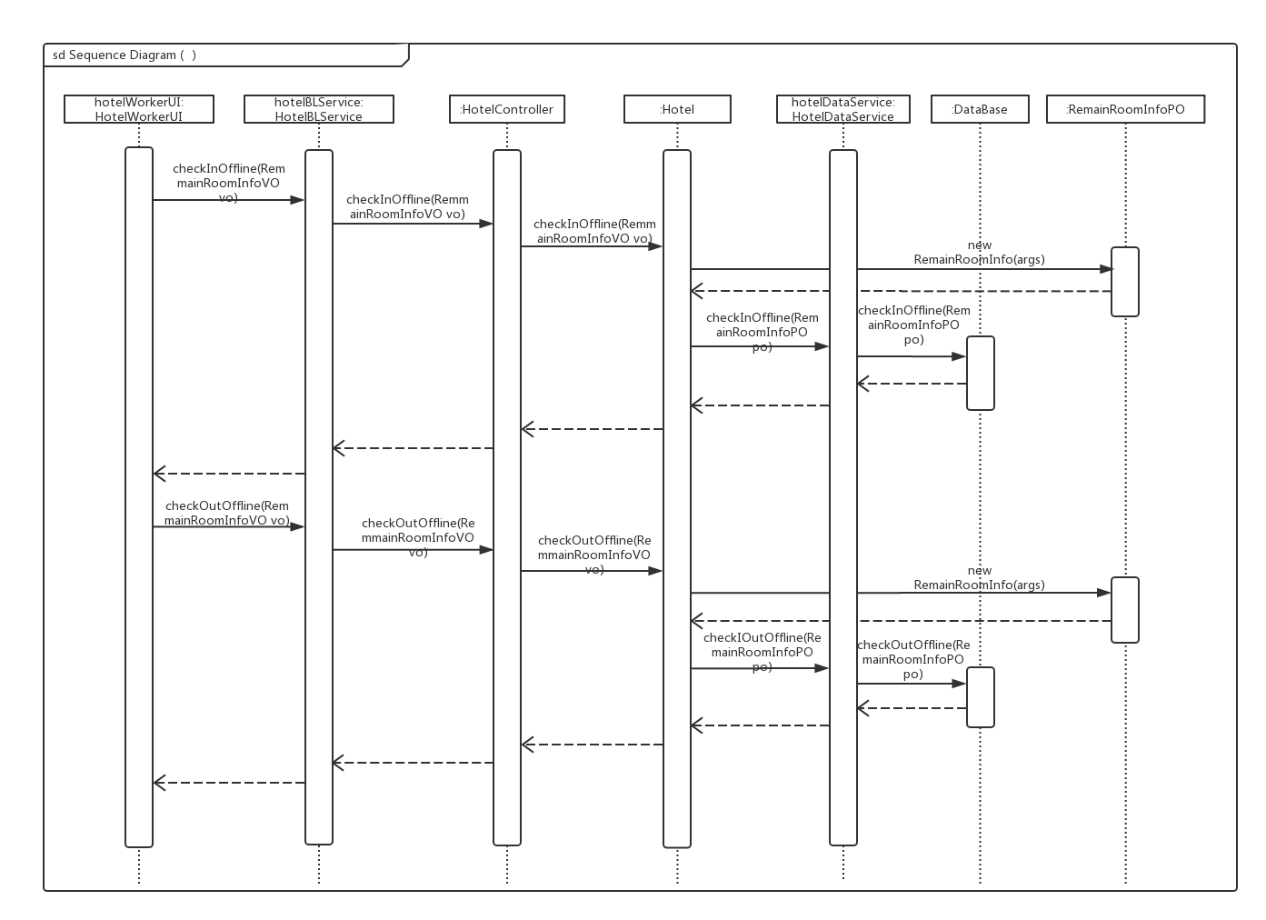
酒店基本信息维护顺序图



客房信息维护顺序图



退房及更新退房信息、剩余客房信息



线下客户入住和退房处理顺序图

1. 业务逻辑层的设计原理

利用委托式控制风格，每个界面需要访问的业务逻辑由各自的控制器委托给不同的领域对象。

### promotion模块

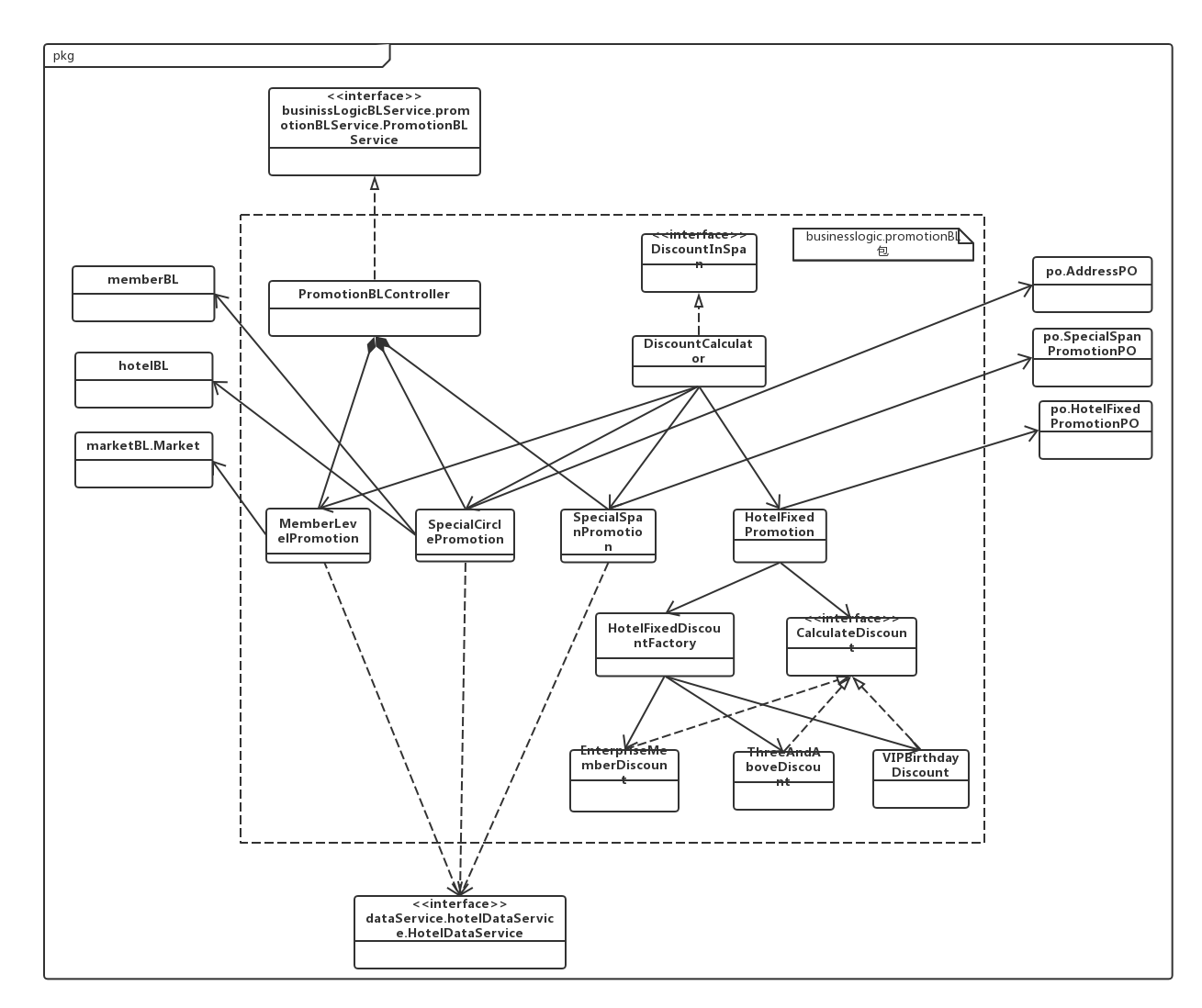
1. 模块概述

promotion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。

promotion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promotionBLService接口。为了隔离业务逻辑职责和逻辑控制职责，添加promotionController，这样promotionController会将对查看及维护促销策略的业务逻辑处理委托给promotion对象。PromotionPO是作为促销策略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） | | |
| promotion.getHotelFixedPromotions | 语法 | Public Iterator<HotelFixedPromotionVO> getHotelFixedPromotions(String hotelID) |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 系统返回该酒店的固定促销策略集合 |
| promotion.updateHotelFixedPromotion | 语法 | public ResultMessage updateHotelFixedPromotion(HotelFixedPromotionVO hotelFixedPromotionVO); |
| 前置条件 | 酒店工作人员修改一条固定策略折扣，请求修改 |
| 后置条件 | 系统更新并保存当前固定策略的折扣，并返回是否成功更新 |
| promotion.getHotelSpecialSpanPromotions | 语法 | public Iterator<SpecialSpanPromotionVO> getHotelSpecialSpanPromotions(String userID); |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 系统返回酒店特定期间促销策略集合 |
| promotion.getWebSpecialSpanPromotions | 语法 | public Iterator<SpecialSpanPromotionVO> getWebSpecialSpanPromotions(); |
| 前置条件 | 网站营销人员请求查看网站特定期间促销策略 |
| 后置条件 | 系统返回网站特定期间促销策略 |
| promotion.addHotelSpecialSpanPromotion | 语法 | public ResultMessage addHotelSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知添加的酒店特定期间策略的信息 |
| 后置条件 | 系统添加并保存一条特定期间策略，并返回是否成功添加 |
| promotion.addWebSpecialSpanPromotion | 语法 | public ResultMessage addWebSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知添加的网站特定期间策略的信息 |
| 后置条件 | 系统添加并保存一条特定期间策略，并返回是否成功添加 |
| promotion.updateHotelSpecialSpanPromotion | 语法 | public ResultMessage updateHotelSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知修改的酒店特定期间策略的信息 |
| 后置条件 | 系统更新并保存该条特定期间策略的信息，返回是否成功更新 |
| promotion.updateWebSpecialSpanPromotion | 语法 | public ResultMessage updateWebSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知修改的网站特定期间策略的信息 |
| 后置条件 | 系统更新并保存该条特定期间策略的信息，返回是否成功更新 |
| promotion.deleteHotelSpecialSpanPromotion | 语法 | public ResultMessage deleteHotelSpecialSpanPromotion(String userID,String promotionName) |
| 前置条件 | 已知酒店编号和被删除的特定期间策略的名称 |
| 后置条件 | 系统删除该特定期间策略，返回是否删除成功 |
| promotion.deleteWebSpecialSpanPromotion | 语法 | public ResultMessage deleteWebSpecialSpanPromotion(String promotionName) |
| 前置条件 | 已知被删除的网站特定期间策略的名称 |
| 后置条件 | 系统删除该条特定期间策略，返回是否删除成功 |
| promotion.getSpecialCirclePromotions | 语法 | public Iterator<AddressVO> getSpecialCirclePromotions(String city) |
| 前置条件 | 已知城市名称 |
| 后置条件 | 系统返回该城市的特定商圈vip促销策略集合 |
| promotion.updateSpecialCirclePromotions | 语法 | public ResultMessage updateSpecialCirclePromotions(AddressVO addressVO) |
| 前置条件 | 已知修改的特定商圈折扣信息 |
| 后置条件 | 系统更新并保存该条修改的特定商圈折扣信息，返回是否成功修改 |
| promotion.getSpecialCirclePromotion | 语法 | public double getSpecialCirclePromotion(String city, String circle) |
| 前置条件 | 已知城市和商圈的名称 |
| 后置条件 | 系统返回该城市商圈的vip折扣 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| hotelFixedPromotion. getHotelFixedPromotion(String hotelWorkerID) | 得到该酒店管理的固定策略列表 | |
| hotelFixedPromotion. updateHotelFixedPromotion(HotelFixedPromotionPO hotelFixedPromotionPO) | 更新一条酒店固定策略，返回是否更新成功 | |
| specialSpanPromotion.getHotelSpecialSpanPromotion(String hotelID) | 得到该酒店管理的特定期间促销策略列表 | |
| specialSpanPromotion. getWebSpecialSpanPromotion() | 得到网站的特定期间促销策略策略列表 | |
| specialSpanPromotion. addSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 添加一条酒店特定期间促销策略，返回是否添加成功 | |
| specialSpanPromotion. addWebSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 添加一条网站特定期间促销策略，返回是否添加成功 | |
| specialSpanPromotion. updateSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 更新一条酒店特定期间促销策略列表，返回是否更新成功 | |
| specialSpanPromotion. updateWebSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 更新一条网站特定期间促销策略列表，返回是否更新成功 | |

1. Domain类的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| hotelFixedPomotion.getHotelFixedPromotions | 语法 | Public Iterator<HotelFixedPromotionVO> getHotelFixedPromotions(String hotelID) |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 系统返回该酒店的固定促销策略集合 |
| hotelFixedPromotion.updateHotelFixedPromotion | 语法 | public ResultMessage updateHotelFixedPromotion(HotelFixedPromotionVO hotelFixedPromotionVO); |
| 前置条件 | 酒店工作人员修改一条固定策略折扣，请求修改 |
| 后置条件 | 系统更新并保存当前固定策略的折扣，并返回是否成功更新 |
| hotelFixedPomotion.getDiscountOneday | 语法 | Public getDiscountOneday(PreOrderVO preOrder, LocalDate today) |
| 前置条件 | 已知预订单以及当天日期 |
| 后置条件 | 系统返回满足当天固定策略的折扣 |
| specialSpanPromotion. getHotelSpecialSpanPromotions | 语法 | public Iterator<SpecialSpanPromotionVO> getHotelSpecialSpanPromotions(String userID); |
| 前置条件 | 已知酒店编号 |
| 后置条件 | 系统返回酒店特定期间促销策略集合 |
| specialSpanPromotion.getWebSpecialSpanPromotions | 语法 | public Iterator<SpecialSpanPromotionVO> getWebSpecialSpanPromotions(); |
| 前置条件 | 网站营销人员请求查看网站特定期间促销策略 |
| 后置条件 | 系统返回网站特定期间促销策略 |
| specialSpanPromotion. addHotelSpecialSpanPromotion | 语法 | public ResultMessage addHotelSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知添加的酒店特定期间策略的信息 |
| 后置条件 | 系统添加并保存一条特定期间策略，并返回是否成功添加 |
| specialSpanPromotion.addWebSpecialSpanPromotion | 语法 | public ResultMessage addWebSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知添加的网站特定期间策略的信息 |
| 后置条件 | 系统添加并保存一条特定期间策略，并返回是否成功添加 |
| specialSpaPromotion.updateHotelSpecialSpanPromotion | 语法 | public ResultMessage updateHotelSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知修改的酒店特定期间策略的信息 |
| 后置条件 | 系统更新并保存该条特定期间策略的信息，返回是否成功更新 |
| specialSpanPromotion. updateWebSpecialSpanPromotion | 语法 | public ResultMessage updateWebSpecialSpanPromotion(SpecialSpanPromotionVO specialSpanPromotionVO) |
| 前置条件 | 已知修改的网站特定期间策略的信息 |
| 后置条件 | 系统更新并保存该条特定期间策略的信息，返回是否成功更新 |
| specialSpanPromotion.deleteHotelSpecialSpanPromotion | 语法 | public ResultMessage deleteHotelSpecialSpanPromotion(String userID,String promotionName) |
| 前置条件 | 已知酒店编号和被删除的特定期间策略的名称 |
| 后置条件 | 系统删除该特定期间策略，返回是否删除成功 |
| specialSpanPromotion.deleteWebSpecialSpanPromotion | 语法 | public ResultMessage deleteWebSpecialSpanPromotion(String promotionName) |
| 前置条件 | 已知被删除的网站特定期间策略的名称 |
| 后置条件 | 系统删除该条特定期间策略，返回是否删除成功 |
| specialSpanPromotion.getDiscountOneday | 语法 | Public getDiscountOneday(String hotelID, LocalDate today) |
| 前置条件 | 已知酒店编号、当天日期 |
| 后置条件 | 返回满足的特定期间折扣 |
| specialCirclePromotion.getSpecialCirclePromotions | 语法 | public Iterator<AddressVO> getSpecialCirclePromotions(String city) |
| 前置条件 | 已知城市名称 |
| 后置条件 | 系统返回该城市的特定商圈vip促销策略集合 |
| specialCirclePromotion.updateSpecialCirclePromotions | 语法 | public ResultMessage updateSpecialCirclePromotions(AddressVO addressVO) |
| 前置条件 | 已知修改的特定商圈折扣信息 |
| 后置条件 | 系统更新并保存该条修改的特定商圈折扣信息，返回是否成功修改 |
| specialCirclePromotion. getSpecialCirclePromotion | 语法 | public double getSpecialCirclePromotion(String city, String circle) |
| 前置条件 | 已知城市和商圈的名称 |
| 后置条件 | 系统返回该城市商圈的vip折扣 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| promotionDataService. getHotelFixedPromotion(String hotelWorkerID) | 得到数据库中对应该酒店管理的固定策略列表 | |
| promotionDataService. updateHotelFixedPromotion(HotelFixedPromotionPO hotelFixedPromotionPO) | 更新数据库中一条酒店固定策略，返回是否更新成功 | |
| promotionDataService.getHotelSpecialSpanPromotion(String hotelID) | 得到数据库中对应该酒店管理的特定期间促销策略列表 | |
| promotionDataService. getWebSpecialSpanPromotion() | 得到数据库中对应网站的特定期间促销策略策略列表 | |
| promotionDataService. getHotelFixedPromotion(String hotelWorkerID) | 得到数据库中对应该酒店管理的固定策略列表 | |
| promotionDataService. addSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 向数据库中添加一条特定期间促销策略，返回是否添加成功 | |
| promotionDataService. updateSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 更新数据库中一条特定期间促销策略列表，返回是否更新成功 | |
| promotionDataService. deleteSpecialSpanPromotion(String userID,String promotionName) | 删除数据库中一条特定期间促销策略，返回是否删除成功 | |
| promotionDataService. getSpecialCirclePromotion(String city) | 返回对应该城市的vip特定商圈促销策略列表 | |
| promotionDataService. updateSepecialCirclePromotion(AddressPO addressPO) | 更新数据库中一条vip特定商圈促销策略，返回是否更新成功 | |
| promotionDataService. getSpecialCircleDiscount(String city, String cycle) | 返回指定城市、商圈的vip特定商圈折扣 | |
| member.isMember([String](eclipse-javadoc:%E2%98%82=ClientSystem/src%5C/main%5C/java%3CbusinessLogic.memberBL%7BMember.java%E2%98%83Member~isMember~QString;~QMemberType;%E2%98%82java.lang.String) userID, [MemberType](eclipse-javadoc:%E2%98%82=ClientSystem/src%5C/main%5C/java%3CbusinessLogic.memberBL%7BMember.java%E2%98%83Member~isMember~QString;~QMemberType;%E2%98%82utilities.enums.MemberType) memberType) | 判断该用户是否为指定会员类型 | |
| member.getMemberInfo([String](eclipse-javadoc:%E2%98%82=ClientSystem/src%5C/main%5C/java%3CbusinessLogic.memberBL%7BMember.java%E2%98%83Member~getMemberInfo~QString;%E2%98%82java.lang.String) userID) | 根据用户编号，返回该用户的普通会员信息 | |
| market.getMemberDiscout([String](eclipse-javadoc:%E2%98%82=ClientSystem/src%5C/main%5C/java%3CbusinessLogic.marketBL%7BMarket.java%E2%98%83Market~getMemberDiscout~QString;%E2%98%82java.lang.String) userID) | 根据用户编号，返回该用户的会员等级折扣 | |
| hotel.getHotelAddress([String](eclipse-javadoc:%E2%98%82=ClientSystem/src%5C/main%5C/java%3CbusinessLogic.hotelBL.hotel%7BHotel.java%E2%98%83Hotel~getHotelAddress~QString;%E2%98%82java.lang.String) hotelID) | 根据酒店编号，返回酒店的地址信息 | |
| promotionDataService. getHotelFixedPromotion(String hotelWorkerID) | 得到数据库中对应该酒店管理的固定策略列表 | |
| promotionDataService. updateHotelFixedPromotion(HotelFixedPromotionPO hotelFixedPromotionPO) | 更新数据库中一条酒店固定策略，返回是否更新成功 | |
| promotionDataService.getHotelSpecialSpanPromotion(String hotelID) | 得到数据库中对应该酒店管理的特定期间促销策略列表 | |
| promotionDataService. getWebSpecialSpanPromotion() | 得到数据库中对应网站的特定期间促销策略策略列表 | |
| promotionDataService. getHotelFixedPromotion(String hotelWorkerID) | 得到数据库中对应该酒店管理的固定策略列表 | |
| promotionDataService. addSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 向数据库中添加一条特定期间促销策略，返回是否添加成功 | |
| promotionDataService. updateSpecialSpanPromotion(SpecialSpanPromotionPO specialSpanPromotionPO) | 更新数据库中一条特定期间促销策略列表，返回是否更新成功 | |
| promotionDataService. deleteSpecialSpanPromotion(String userID,String promotionName) | 删除数据库中一条特定期间促销策略，返回是否删除成功 | |
| promotionDataService. getSpecialCirclePromotion(String city) | 返回对应该城市的vip特定商圈促销策略列表 | |
| promotionDataService. updateSepecialCirclePromotion(AddressPO addressPO) | 更新数据库中一条vip特定商圈促销策略，返回是否更新成功 | |
| promotionDataService. getSpecialCircleDiscount(String city, String cycle) | 返回指定城市、商圈的vip特定商圈折扣 | |

1. 业务逻辑层的动态模型



酒店促销策略维护顺序图

1. 业务逻辑层的设计原理

利用委托式控制风格，每个界面需要访问的业务逻辑由各自的控制器委托给不同的领域对象。