### 业务逻辑模块的接口模范

1. **logIn模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| logIn.guestSignUp | 语法 | GuestVO guestSignUp (UserVO 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 |
| logIn.logIn | 语法 | UserType logIn (String userID, String password) |
| 前置条件 | 用户输入的用户编号和密码符合规范 |
| 后置条件 | 用户被认证并进入其相应的主界面 |
| logIn.logOut | 语法 | void logOut (String webManager) |
| 前置条件 | 无 |
| 后置条件 | 系统退出，进入登录主界面 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| User.add(UserVo newUserVo, UserType userType) | 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. **order模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| order.getAllOrderGenerals | 语法 | Iterator<OrderGeneralVO> getAllOrderGenerals(String userID, UserType userType) |
| 前置条件 | 客户／酒店工作人员已经登录 |
| 后置条件 | 无 |
| order.getSpecialOrderGenerals | 语法 | Iterator<OrderGeneralVO> getSpecialOrderGenerals(String userID, UserType userType, OrderState orderState) |
| 前置条件 | 客户／酒店工作人员已经登录 |
| 后置条件 | 无 |
| order.getOrderDetail | 语法 | OrderVO getOrderDetail(String orderID) |
| 前置条件 | 客户／酒店工作人员／网站营销人员已经登录 |
| 后置条件 | 无 |
| order.getEvaluations | 语法 | Iterator<HotelEvaluationVO> getEvaluations(String hotelID) |
| 前置条件 | 客户／酒店工作人员已经登录 |
| 后置条件 | 无 |
| order.getCalculatedPrice | 语法 | int getCalculatedPrice(PreOrderVO preOrderVO) |
| 前置条件 | 客户已经登录，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.createOrder | 语法 | ResultMessage createOrder(OrderVO orderVO) |
| 前置条件 | 客户已经登录，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.undoNormalOrder | 语法 | ResultMessage undoNormalOrder(String orderID) |
| 前置条件 | 客户已经登录，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.addEvaluation | 语法 | ResultMessage addEvaluation(GuestEvaluationVO evaluationVO) |
| 前置条件 | 客户已经登录，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.getAllGuestCommentOrderGeneral | 语法 | Iterator<OrderGeneralVO> getAllGuestCommentOrderGeneral(String guestID, boolean hasCommented) |
| 前置条件 | 客户已经登录 |
| 后置条件 | 无 |
| order.getMyOrdersOfThisHotel | 语法 | Iterator<OrderGeneralVO> getMyOrdersOfThisHotel(String guestID, String hotelID) |
| 前置条件 | 客户已经登录 |
| 后置条件 | 无 |
| order.updateCheckIn | 语法 | ResultMessage updateCheckIn(CheckInVO checkInVO) |
| 前置条件 | 酒店工作人员已经登录 |
| 后置条件 | 系统更改此订单状态为已执行订单并持久化保存该客户的订单数据 |
| order.updateCheckOut | 语法 | ResultMessage updateCheckOut(CheckOutVO checkOutVO) |
| 前置条件 | 酒店工作人员已经登录 |
| 后置条件 | 系统更改此订单状态和酒店剩余房间数量，并持久化保存该客户的订单数据 |
| order.getAllHotelCheckOutOrderGeneral | 语法 | Iterator<OrderGeneralVO> getAllHotelCheckOutOrderGeneral(String hotelID, boolean hasCheckOut) |
| 前置条件 | 酒店工作人员已经登录 |
| 后置条件 | 无 |
| order.undoAbnormalOrder | 语法 | ResultMessage undoAbnormalOrder(String orderID, double percent) |
| 前置条件 | 网站营销人员已登录，订单状态为异常 |
| 后置条件 | 系统更改订单状态为已撤销订单并持久化保存该客户的订单数据 |
| order.getAllAbnormalOrderGeneral | 语法 | List<OrderGeneralVO> getAllAbnormalOrderGeneral(LocalDate date) |
| 前置条件 | 网站营销人员已经登录 |
| 后置条件 | 无 |
| order.getAllUnexecutedOrderGeneral | 语法 | List<OrderGeneralVO> getAllUnexecutedOrderGeneral(LocalDate date) |
| 前置条件 | 网站营销人员已经登录 |
| 后置条件 | 无 |
| order.getBookedHotel | 语法 | List<String> getBookedHotels(String guestID) |
| 前置条件 | 无 |
| 后置条件 | 无 |
| order.getOrderState | 语法 | OrderState getOrderState(String guestID, String hotelID) |
| 前置条件 | 无 |
| 后置条件 | 无 |
| 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.getAllAbnormalOrderGeneral (Date date) | 根据ID查找持久化对象 | |
| OrderDataService.getAllAbnormalOrderGeneral () | 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 | |
| User.getSingle(String userID,UserType userType) | 获得单一客户的持久化对象信息 | |
| Promotion.getDiscount(PreOrder preOrder) | 获得单一对象 | |

1. **hotel模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| hotel.getHotelList | 语法 | Public List<HotelGeneralVO> getHotelList(AddressVO addressVO) |
| 前置条件 | 已知地址和商圈的信息，需要浏览商圈内所有酒店的概况 |
| 后置条件 | 系统根据地址和商圈返回符合要求的酒店信息列表 |
| hotel.getSortedHotels | 语法 | Public List<HotelGeneralVO> getSortedHotels(SortStrategy sortStrategy) |
| 前置条件 | 已知该商圈的所有酒店列表以及排序的规则（价格，星级，评分），需要对当前酒店列表按照特定的规则排序 |
| 后置条件 | 系统返回以排序规则排好序的酒店列表 |
| hotel.getBookedHotels | 语法 | Public List<HotelGeneralVO> getBookedHotels(String userID) |
| 前置条件 | 已知该商圈的所有酒店列表以及客户的编号，需要查看该客户已经预定过的酒店 |
| 后置条件 | 系统根据客户的编号，返回该客户已经预定过的酒店列表 |
| hotel.getHotelDetail | 语法 | Public HotelVO  getHotelDetail(String hotelID) |
| 前置条件 | 客户需要查看酒店的详细信息，已知酒店的编号 |
| 后置条件 | 系统返回该酒店的详细信息 |
| hotel.getUncommentedHotels | 语法 | Public List<HotelGeneralVO> getUncommentedHotels(String userID) |
| 前置条件 | 客户查看已执行订单并要评价还未评价的酒店，已知客户的编号 |
| 后置条件 | 返回客户所有已执行订单的酒店列表 |
| hotel.updateEvaluation | 语法 | Public ResultMessage update  Evaluation(EvaluationVO evaluationVO) |
| 前置条件 | 已知客户当前评价的酒店信息表以及客户选定的评分等级和输入的评论，客户确认更新评论 |
| 后置条件 | 系统更新并持久化保存当前酒店评价信息表，并更新酒店评分值，系统提示客户评论是否成功 |
| hotel.getHotelInfo | 语法 | Public HotelVO getHotelInfo(String userID) |
| 前置条件 | 已知酒店工作人员的编号，需要维护酒店基本信息 |
| 后置条件 | 系统返回该酒店工作人员对应的酒店基本信息 |
| hotel.updateHotelInfo | 语法 | Public ResultMessage updateHotelInfo (HotelVO hotelVO) |
| 前置条件 | 酒店工作人员确认修改酒店基本信息，已知修改后的酒店基本面信息 |
| 后置条件 | 系统更新并持久化保存酒店基本信息，并提示酒店工作人员保存是否成功 |
| hotel.getHotelRoomInfo | 语法 | Public RoomInfoVO getHotelRoomInfo (String userID) |
| 前置条件 | 已知酒店工作人员的编号，需要维护客房信息 |
| 后置条件 | 系统返回该酒店工作人员管理的客房信息 |
| hotel.updateHotelRoomInfo | 语法 | Public ResultMessage updateHotelRoomInfo (RoomInfoVO RoomInfoVO) |
| 前置条件 | 酒店工作人员确认修改客房信息，已知修改后的客房信息 |
| 后置条件 | 系统更新并持久化保存客房信息，并提示酒店工作人员保存是否成功 |
| hotel.updateCheckIn | 语法 | Public ResultMessage updateCheckIn (List<String> roomNum,Calendar inTime,Calendar predictedOutTime) |
| 前置条件 | 酒店工作人员已执行订单，并已知入住房间号的列表，入住时间以及预计离开时间 |
| 后置条件 | 系统更新并持久化保存入住信息，并提示酒店工作人员是否更新成功 |
| hotel.updateCheckOut | 语法 | Public ResultMessage updateCheckOut (Calendar outTime) |
| 前置条件 | 酒店工作人员对订单办理退房，已知退房时间 |
| 后置条件 | 系统更新并持久化保存退房信息，并提示酒店工作人员是否更新成功 |
| hotel.getRemainRoomInfo | 语法 | Public RemainRoomInfoVO getRemainRoomInfo (String userID) |
| 前置条件 | 已知酒店工作人员的编号，需要对线下客户的入住和退房进行处理或需要查看剩余客房信息 |
| 后置条件 | 系统返回该酒店工作人员所对应酒店的剩余客房信息 |
| hotel.updateRemainRoomInfo | 语法 | Public ResultMessage updateRemainRoomInfo (Operation operation, Map<RoomType, Integer> roomInfo) |
| 前置条件 | 在线下客户入住和退房处理中, 酒店工作人员确认选择的操作及输入的房型和该房型的数量或者酒店工作人员已执行订单以及为已执行订单退房 |
| 后置条件 | 系统更新并持久化保存酒店剩余客房信息, 并提示酒店工作人员是否保存成功 |
| hotel.add | 语法 | Public ResultMessage add (HotelVO hotelVO) |
| 前置条件 | 网站管理人员确认添加酒店, 已知添加的酒店信息 |
| 后置条件 | 系统更新并持久化保存添加的酒店信息 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| hotelDataService.getHotelList(Address VO addressVO) | 得到数据库里的当前商圈、地址信息里的所有酒店列表 | |
| hotelDataService.getHotelDetail(String hotelID) | 得到数据库里的对应当前酒店编号的详细信息 | |
| hotelDataService.addEvaluation(EvaluationPO evaluationPO) | 在数据库里给对应当前酒店编号的酒店添加评论 | |
| hotelDataService.getHotelInfo(String hotelID) | 从数据库里获得本酒店的酒店基本信息 | |
| hotelDataService.setHotelInfo(HotelInfoPO  hotelInfoPO) | 修改数据库里本酒店酒店基本信息 | |
| hotelDataService.getHotelRoomInfo(String hotelID) | 从数据库里获得本酒店的客房信息 | |
| hotelDataService.setHotelRoomInfo (HotelRoomInfoPO hotelRoomInfoPO) | 修改数据库里本酒店的客房信息 | |
| hotelDataService.setCheckInInfo(CheckInInfo checkInInfo) | 修改数据库里订单的入住信息 | |
| hotelDataService.setCheckOutInfo(CheckOutInfo checkOutInfo) | 修改数据库里该订单的退房时间 | |
| hotelDataService.getRemainRoomInfo(String hotelID) | 获取数据库里该酒店的剩余房间信息 | |
| hotelDataService.setRemainRoomInfo  (RemainRoomInfoPO remainRoomInfoPO) | 修改数据库里该酒店的剩余房间信息 | |
| hotelDataService.add(HotelInfoPO hotelInfoPO) | 在数据库里添加一条就酒店的信息 | |
| promotion.getPromotions(String userID) | 依据酒店工作人员编号得到数据库里当前酒店的所有策略信息列表 | |
| order.getBookedHotels(String guestID) | 从所有订单中得到该用户预定过的酒店 | |

1. **promotion模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| promotion.getPromotions | 语法 | Public List<promotionVO> getPromotions (String userID) |
| 前置条件 | 已知用户编号（可以是酒店工作人员，也可以是网站营销人员），需要维护促销策略 |
| 后置条件 | 系统返回当前用户所管理的促销策略的列表 |
| promotion.update | 语法 | Public ResultMessage update (List<PromotionVO> list) |
| 前置条件 | 用户（酒店工作人员或网站营销人员）确认更新促销策略，已知当前的促销策略的列表 |
| 后置条件 | 更新并持久化保存当前促销策略，并提示用户更新是否成功 |
| promotion.getDiscount | 语法 | Public float getDiscout(PreOrder preOrder) |
| 前置条件 | 客户在填写订单时，订单需要知道当前订单所能使用的最大折扣率，已知该预定酒店的编号，商圈和地址信息，以及该客户是否为会员 |
| 后置条件 | 系统返回当前订单所能使用的最大折扣率 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| promotionDataService.  getPromotions(String userID) | 得到数据库中对应用户所管理的策略列表 | |
| promotionDataService.  setPromotions(String userID,List<promotionPO> list) | 修改数据库中对应用户所管理的策略列表 | |

1. **credit模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| credit.charge | 语法 | Public ResultMessage charge(int chargeNum) |
| 前置条件 | 网站营销人**员**为客户信用充值 |
| 后置条件 | 系统为客户信用值充值并返回是否成功 |
| credit.getBasicInfo | 语法 | Public CreditVO getBasicInfo (String ID) |
| 前置条件 | 用户（酒店工作人员或网站营销人员）确认更新促销策略，已知当前的促销策略的列表 |
| 后置条件 | 系统返回与该ID对应的包含信用值及信用记录的客户个人基本信息 |
| 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对应的用户信息 | |
| User.modifyCredit(float creditNum) | 返回客户信用值是否修改成功 | |
| market.getMemberFormulation() | 返回现有的会员等级制度信息 | |
| order.getAllGuestOrderGeneral (String guestID) | 返回与该ID对应的客户所有订单概况 | |

1. **member模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| Memeber.add | 语法 | Public ResultMessage add(MemberVO memberVO) |
| 前置条件 | 会员信息符合规则且齐全 |
| 后置条件 | 系统根据已指定会员类型持久化为该客户增加会员数据 |
| Member.modify | 语法 | Public ResultMessage modify(MemberVO memberVO) |
| 前置条件 | 会员信息符合规则且齐全 |
| 后置条件 | 系统修改该会员并持久化保存该客户的会员数据 |
| Member.getMemberInfo | 语法 | Public MemberVO getMemberInfo(String userID,MemberType memberType) |
| 前置条件 | userID.Length=对应用户类型的ID长度 |
| 后置条件 | 系统返回与该userID对应的指定会员类型 |
| Member.isMember | 语法 | public boolean isMember(String userID, MemberType memberType) |
| 前置条件 | userID.Length=对应用户类型的ID长度 |
| 后置条件 | 返回该userID的用户是否为会员 |
| Member.getMemberType | 语法 | public MemberType getMemberType(String userID, MemberType memberType) |
| 前置条件 | userID.Length=对应用户类型的ID长度 |
| 后置条件 | 系统返回该userID的会员类型 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| GuestDataService.ModifyMember(MemberPO memberPO) | 根据提供的memberPO修改单一持久化客户对象的信用值 | |
| GuestDataService.  getSingle (String userID) | 根据ID查找客户持久化对象 | |

1. **user模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| User.getAll | 语法 | Public List<UserVO> getAll(UserType userType) |
| 前置条件 | 无 |
| 后置条件 | 系统返回指定用户类型的所有用户信息 |
| User.getSingle | 语法 | Public UserVO getSingle(String userID，UserType userType) |
| 前置条件 | userID.Length=对应用户类型的ID长度 |
| 后置条件 | 系统返回与该ID对应的user信息 |
| User.add | 语法 | Public ResultMessage add(UserVO newUserVO) |
| 前置条件 | 输入的userID不存在，password和所有信息符合规范 |
| 后置条件 | 系统新增该用户并持久化保存该用户的数据 |
| User.addHotel | 语法 | Public ResultMessage addHotel(HotelVO newHotelVO, String hotelID) |
| 前置条件 | 输入的hotelID不存在,所有酒店信息符合规范 |
| 后置条件 | 系统新增该酒店并持久化保存该酒店的数据 |
| User.modify | 语法 | Public ResultMessage modify(UserVO userVO) |
| 前置条件 | 用户信息符合规则且齐全 |
| 后置条件 | 系统修改该用户信息并持久化保存该用户数据 |
| User.modifyCredit | 语法 | Public ResultMessage modifyCredit (float creditNum) |
| 前置条件 | 网站营销人员为客户信用充值 |
| 后置条件 | 系统修改该客户信用值并持久化保存该用户数据 |
| User.getLoginInfo | 语法 | Public ResultMessage getLoginInfo(String userID, UserType userType) |
| 前置条件 | userID.Length=对应用户类型的ID长度 |
| 后置条件 | 系统返回与该用户ID对应的密码  如果ID不存在，则系统返回该ID对应用户不存在 |
|  | | |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| guestDataService.  add(GuestPO newGuestPO) | 根据提供的newGuestPO增加客户持久化对象 | |
| guestDataService.  getSingle (String guestID) | 根据userID查找客户持久化对象 | |
| guestDataService.modify  (GuestPO guestPO) | 修改单一客户持久化对象 | |
| guestDataService.getAll() | 获取所有客户持久化对象 | |
| hotelWorkerDataService.  add(HotelWorkerPO newHotelWorkerPO) | 根据提供的newHotelWorkerPO增加酒店工作人员持久化对象 | |
| hotelWorkerDataService.  getSingle (String hotelWorkerID) | 根据userID查找酒店工作人员持久化对象 | |
| hotelWorkerDataService.modify  (HotelPO hotelPO) | 修改单一酒店工作人员持久化对象 | |
| hotelWorkerDataService.getAll() | 获取所有酒店工作人员持久化对象 | |
| webMarketerDataService.  add(WebMarketerPO newWebMarketerPO) | 根据提供的newWebMarkterPO增加网站营销人员持久化对象 | |
| webMarkterDataService.  getSingle (String webMarketerID) | 根据userID查找网站营销人员持久化对象 | |
| webMarkterDataService.modify(WebMarketerPO webMarketerPO) | 修改单一网站营销人员持久化对象 | |
| webMarketerDataService.getAll() | 获取所有网站营销人员持久化对象 | |
| webManagerDataService.  add(WebManagerPO newWebManagerPO) | 根据提供的newWebManagerPO增加网站管理人员持久化对象 | |
| webManagerDataService.  getSingle (String webManagerID) | 根据userID查找网站管理人员持久化对象 | |
| webManagerDataService.modify  (WebManagerPO webManagerPO) | 修改单一网站管理人员持久化对象 | |
| webManagerDataService.getAll() | 获取所有网站管理人员持久化对象 | |
| Hotel.add | 新增该酒店并持久化保存该酒店的数据 | |

1. **market模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| market.getMemberFormulation | 语法 | Public MarketVO getMemberFormulation () |
| 前置条件 | 无 |
| 后置条件 | 返回现有的会员等级制度信息 |
| market.setMemberFormulation | 语法 | Public ResultMessage setMemberFormulation (MarketVO marketVO) |
| 前置条件 | 网站营销人员输入会员等级制度 |
| 后置条件 | 系统记录此会员等级制度信息 |
| 需要的接口（需接口） | | |
| 服务名 | **服务内容** | |
| marketDataService.getMemberFormulation | 返回现有的会员等级制度信息 | |
| marketDataService.setMemberFormulation(MarketVO marketVO) | 修改会员等级制度信息持久化对象，系统返回是否成功记录 | |

## 数据层的分解

数据层主要给业务逻辑层提供数据访问服务，包括对于持久化数据的增、删、改、查。例如，AccountBusinessService的服务主要由AccountDataService提供。由于持久化数据的保存可能存在多种形式：Txt文件、序列化文件、数据库等，所示抽象了数据服务。以AccountDataService为例，数据层模块的具体描述下图所示：

### 数据层模块的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| orderData | 负责保存订单信息，进行增、改、查的操作 |
| hotelData | 负责保存酒店信息，进行增、改、查的操作 |
| promotionData | 负责保存促销策略信息，进行增、删、改、查的操作 |
| marketData | 负责保存会员等级制度信息，进行改、查的操作 |
| guestData | 负责保存客户信息，进行增、改、查的操作 |
| hotelWorkerData | 负责保存酒店工作人员信息，进行增、改、查的操作 |
| webMarketerData | 负责保存网站营销人员信息，进行改、查的操作 |
| webManagerData | 负责保存网站管理人员信息，进行改、查的操作 |

### 数据层模块的接口规范

1. **orderData模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| orderData.createOrder | 语法 | Public ResultMessage createOrder (OrderPO order) |
| 前置条件 | 不存在相同的orderId |
| 后置条件 | 数据库中添加一条订单数据 |
| orderData.getOrderDetails | 语法 | Public OrderPO getOrderDetail (String orderID) |
| 前置条件 | 无 |
| 后置条件 | 系统返回对应id的OrderPO,如果id不存在，返回Null |
| orderData.getAllGuestOrderGeneral | 语法 | Public List<OrderGeneralPO> getAllGuestOrderGeneral (GuestID GuestID) |
| 前置条件 | 无 |
| 后置条件 | 系统返回对应id的OrderPO,如果id不存在，返回Null |
| orderData. getAllHotelOrderGeneral | 语法 | Public List<OrderGeneralPO> getAllHotelOrderGeneral (GuestID GuestID) |
| 前置条件 | 无 |
| 后置条件 | 系统返回对应id的OrderPO,如果id不存在，返回Null |
| orderData. executeOrder | 语法 | Public ResultMessage executeOrder (String orderID) |
| 前置条件 | 存在相同的orderId且订单状态为未执行 |
| 后置条件 | 数据库中更改未执行订单的状态为已执行 |
| orderData. undoAbnormalOrder | 语法 | Public ResultMessage undoAbnormalOrder (String orderID) |
| 前置条件 | 存在相同的orderId且订单状态为异常 |
| 后置条件 | 数据库中更改并保存订单的状态数据 |
| orderData. undoNormalOrder | 语法 | Public ResultMessage undoNormalOrder (String orderID) |
| 前置条件 | 存在相同的orderId且订单状态为未执行 |
| 后置条件 | 数据库中更改并保存订单的状态 |

1. **hotelData模块的接口规范**

|  |  |  |
| --- | --- | --- |
|  | | |
| hotelData.getHotelList | 语法 | Public List<HotelGeneralPO> getHotelList(AddressPO addressPO) |
| 前置条件 | 已知商圈和地址信息 |
| 后置条件 | 系统在数据库中查询并返回符合该商圈和地址信息的所有酒店列表 |
| hotelData.getHotelDetail | 语法 | Public HotelPO getHotelDetail(String hotelID) |
| 前置条件 | 已知客户点击查看的酒店编号 |
| 后置条件 | 返回保存酒店详细信息的HotelPO对象 |
| hotelData.addEvaluation | 语法 | Public boolean addEvaluation (EvaluationPO evaluationPO) |
| 前置条件 | 已知评论信息的对象evaluationPO以及被评论的酒店编号 |
| 后置条件 | 系统向数据库中添加evaluationPO，并返回添加是否成功的布尔值 |
| hotelData.getHotelInfo | 语法 | Public HotelInfoPO getHotelInfo (String hotelID) |
| 前置条件 | 已知酒店编号，逻辑层想获取酒店的基本信息 |
| 后置条件 | 返回一个保存有该酒店基本信息的HotelInfoPO对象 |
| hotelData.setHotelInfo | 语法 | Public boolean  setHotelInfo(HotelInfoPO hotelInfoPO) |
| 前置条件 | 已知保存有酒店基本信息的HotelInfoPO对象，逻辑层请求修改该酒店的基本信息 |
| 后置条件 | 系统在数据库中更新该HotelInfoPO对象，并返回一个修改是否成功的布尔值 |
| hotelData.getHotelRoomInfo | 语法 | Public HotelRoomInfoPO getHotelRoomInfo(String hotelID) |
| 前置条件 | 已知酒店编号，逻辑层请求查看该酒店的客房信息 |
| 后置条件 | 系统返回一个保存有酒店客房信息的HotelRoomInfoPO的对象 |
| hotelData.setHotelRoomInfo | 语法 | Public boolean setHotelRoomInfo  (HotelRoomInfoPO hotelRoomInfoPO) |
| 前置条件 | 已知一个保存有酒店客房信息的HotelRoomInfoPO对象，逻辑层请求更新数据库的酒店客房信息 |
| 后置条件 | 系统更新数据库里的HotelRoomInfoPO对象，并返回是否修改成功的布尔值 |
| hotelData.setCheckInInfo | 语法 | Public boolean setCheckInInfo (CheckInInfo checkInInfo) |
| 前置条件 | 已知保存有入住的信息的对象，该订单已被执行，需要修改数据库中的入住信息 |
| 后置条件 | 系统修改数据库中的入住信息，并返回一个修改是否成功的布尔值 |
| hotelData.setCheckOutInfo | 语法 | Public boolean setCheckOutInfo (CheckOutInfo checkOutInfo) |
| 前置条件 | 已知保存有退房信息的对象，该客户已办理退房，需要更新数据库里的退房信息 |
| 后置条件 | 系统更新数据库里的退房信息，并返回更新是否成功的布尔值 |
| hotelData.getRemainRoomInfo | 语法 | Public RemainRoomInfoPO getRemainRoomInfo(String hotelID) |
| 前置条件 | 已知酒店的编号，逻辑层请求当前酒店剩余客房的信息 |
| 后置条件 | 系统返回一个保存剩余客房信息的RemainRoomInfoPO对象 |
| hotelData.setRemainRoomInfo | 语法 | Public boolean setRemainRoomInfo  (RemainRoomInfoPO remainRoomInfoPO) |
| 前置条件 | 已知保存有已被修改剩余客房信息的RemainRoomInfo的对象以及当前酒店的编号，逻辑层请求修改此酒店的剩余客房信息 |
| 后置条件 | 系统更新数据库里此酒店的剩余客房信息，并返回一个是否修改成功的布尔值 |
| hotelData.add(HotelInfoPO hotelInfoPO) | 语法 | Public boolean add (HotelInfoPO hotelInfoPO) |
| 前置条件 | 已知需要被添加酒店的HotelInfoPO对象 |
| 后置条件 | 系统向数据库里添加此HotelInfoPO对象，并返回一个添加是否成功的布尔值 |

1. **promotionData模块的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| promotionData.getPromotions | 语法 | Public List<PromotionPO> getPromotions(String userID) |
| 前置条件 | 已知用户编号，逻辑层请求获得该用户（酒店工作人员、网站营销人员）所管理的促销策略 |
| 后置条件 | 返回该用户所管理的促销策略 |
| promotionData.setPromotions | 语法 | Public boolean setPromotions(String userID,List<PromotionPO> list) |
| 前置条件 | 已知用户编号和被修改的促销策略列表，逻辑层请求更新数据库中的策略列表 |
| 后置条件 | 系统更新数据库中的策略列表，并返回一个是否修改成功的布尔值 |

1. **marketData模块的接口规范**

|  |  |  |
| --- | --- | --- |
| marketData.getMemberFormulation | 语法 | Public MarketPO getMemberFormulation() |
| 前置条件 | 无 |
| 后置条件 | 返回现有的会员等级制度信息 |
| marketData.setMemberFormulation | 语法 | Public MessageResult setMemberFormulation (HotelInfoPO hotelInfoPO) |
| 前置条件 | 网站营销人员输入会员等级制度信息 |
| 后置条件 | 系统记录此会员等级制度信息 |

1. **guestData模块的接口规范**

|  |  |  |
| --- | --- | --- |
| guestData.modifyMember | 语法 | Public ResultMessage modifyMember(MemberPO memberPO) |
| 前置条件 | 无 |
| 后置条件 | 根据提供的memberPO修改单一持久化客户对象的信用值 |
| guestData.getSingle | 语法 | Public ResultMessage getSingle (String guestID) |
| 前置条件 | guestID.Length=客户的ID长度10 |
| 后置条件 | 系统返回与该ID对应的客户信息 |
| guestData.add | 语法 | Public ResultMessage add(GuestPO newGuestPO) |
| 前置条件 | 输入的guestID不存在，password和所有客户信息符合规范 |
| 后置条件 | 系统新增该客户并持久化保存该客户的数据 |
| guestData.modify | 语法 | Public ResultMessage modify  (GuestPO guestPO) |
| 前置条件 | 客户信息符合规则且齐全 |
| 后置条件 | 系统修改该用户信息并持久化保存该用户数据 |
| guestData.init | 语法 | Public ResultMessage init() |
| 前置条件 | 无 |
| 后置条件 | 清空所有客户持久化对象 |

1. **hotelWorker模块的接口规范**

|  |  |  |
| --- | --- | --- |
| hotelWorkerData.getSingle | 语法 | Public ResultMessage getSingle (String hotelWorkerID) |
| 前置条件 | hotelWorkerID.Length=酒店工作人员的ID长度8 |
| 后置条件 | 系统返回与该ID对应的酒店工作人员信息 |
| hotelWorkerData.add | 语法 | Public ResultMessage add(HotelWorkerPO newHotelWorkerPO) |
| 前置条件 | 输入的hotelWorkerID不存在，password和所有酒店工作人员信息符合规范 |
| 后置条件 | 系统新增该酒店工作人员并持久化保存该酒店工作人员的数据 |
| hotelWorkerData.modify | 语法 | Public ResultMessage modify  (HotelWorkerPO hotelWorkerPO) |
| 前置条件 | 酒店工作人员信息符合规则且齐全 |
| 后置条件 | 系统修改该酒店工作人员信息并持久化保存该酒店工作人员数据 |
| hotelWorkerData.init | 语法 | Public ResultMessage init() |
| 前置条件 | 无 |
| 后置条件 | 清空所有酒店工作人员持久化对象 |

1. **webMarketer模块的接口规范**

|  |  |  |
| --- | --- | --- |
| webMarketerData.getSingle | 语法 | Public ResultMessage getSingle (String webMarketID) |
| 前置条件 | webMarketID.Length=网站营销人员的ID长度6 |
| 后置条件 | 系统返回与该ID对应的网站营销人员信息 |
| webMarketerData.add | 语法 | Public ResultMessage add(WebMarketerPO newWebMarketerPO) |
| 前置条件 | 输入的webMarketerID不存在，password和所有网站营销人员信息符合规范 |
| 后置条件 | 系统新增该网站营销人员并持久化保存该网站营销人员的数据 |
| webMarketerData.modify | 语法 | Public ResultMessage modify  (WebMarketerPO webMarketerPO) |
| 前置条件 | 网站营销人员信息符合规则且齐全 |
| 后置条件 | 系统修改该网站营销人员信息并持久化保存该网站营销人员数据 |
| webMarketerData.init | 语法 | Public ResultMessage init() |
| 前置条件 | 无 |
| 后置条件 | 清空所有网站营销人员持久化对象 |

1. **webManager模块的接口规范**

|  |  |  |
| --- | --- | --- |
| webManagerData.getSingle | 语法 | Public ResultMessage getSingle (String webManagerID) |
| 前置条件 | webManagerID.Length=网站管理人员的ID长度4 |
| 后置条件 | 系统返回与该ID对应的网站管理人员信息 |
| webManagerData.add | 语法 | Public ResultMessage add(WebManagerPO newWebManagerPO) |
| 前置条件 | 输入的webManagerID不存在，password和所有网站管理人员信息符合规范 |
| 后置条件 | 系统新增该网站管理人员并持久化保存该网站管理人员的数据 |
| webManagerData.modify | 语法 | Public ResultMessage modify  (WebManagerPO webManagerPO) |
| 前置条件 | 网站管理人员信息符合规则且齐全 |
| 后置条件 | 系统修改该网站管理人员信息并持久化保存该网站管理人员数据 |