## 11/25

修改ManageOrder. checkIn&checkOut方法参数

修改OrderData. searchByUser&searchByHotel方法参数

删除OrderData. hasBeenOrdered方法

## 11/26

修改HotelBriefVO成员

## 11/27

用户名为6-16位字符串，密码为6-16位字符串

## 11/28

OrderPO, OrderVO增加订单价格。

修改ManageUser. searchHotelStaff的返回类型：UserVO



修改SearchHotel. getHotelList方法

的参数类型：HotelSort

## 11/30

用户名和密码为6-16个字符组成的字符串，用户名仅包含英文字母、数字或下划线。

OrderVO、OrderPO增加了酒店名称。

OrderFilterVO增加了酒店编号。酒店工作人员查看订单时使用酒店编号属性，客户查看订单时则使用酒店名称属性。

String类型的参数的初始值（不需要这个参数时的值）为null。

OrderVO、OrderPO增加了订房者真实姓名。

CreateOrder. cancelPreOrder & getStrategy的参数修改为int userID



日期标准格式为"yyyy.MM.dd hh-mm-ss"

UserData. getCreditBalance方法返回的信用值以字符串形式存放在ResultMessage的message属性中，isSuccessful表示是否根据参数userID找到了相应的用户。

## 12/1

HotelDetailsVO增加了用户订单列表成员。

## 12/2

RoomPO/VO, RoomTypePO/VO, ReviewPO/VO添加酒店编号（int类型）

VO/PO中整数/浮点数默认值为-1。

PreOrderVO中增加用户名、订房者真实姓名。

OrderInfoVO删去预订编号。

ManageUser. addCreditItem返回类型改为ResultMessage。

PreOrder、cancelPreOrder方法从RoomDataService移到OrderDataService。

RoomDataService. insertType&updateType&deleteType删去参数int hotelID。

增加RoomDataService. getRoomList方法

OrderInfoVO增加订单编号（预订的订单）

OrderVO/PO的房间编号成员改为房间编号与房号表，即HashMap<Integer, String>，key为房间编号，String为实际房号。

线下入住的订单由用户编号为-1标记。

## 12/3

UserPO/VO增加是否已登录（boolean）

UserDataService. searchUser方法增加UserType参数代表用户类型。

增加UserDataService. hotelStaffList & marketerList方法。



HotelBriefVO增加酒店介绍。

HotelDetailsVO增加评论列表。

## 12/4

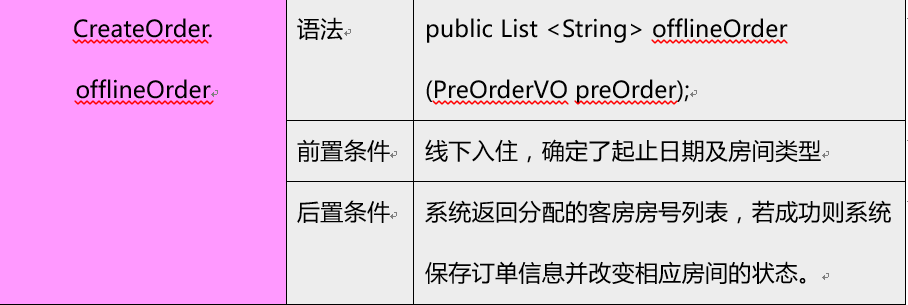
ManageUser. accurateSearch方法仅保留String username。

UserDataService增加UserPO accurateSearch (String username)方法。

OrderPO/VO增加订单生成日期成员（String orderTime）。

PreOrderVO增加客房单价。

增加CreateOrder. offlineOrder方法。



availableRoomType除了第一个hotelID以外其他全是-1、null（默认值）的时候就返回该酒店的房间类型列表。

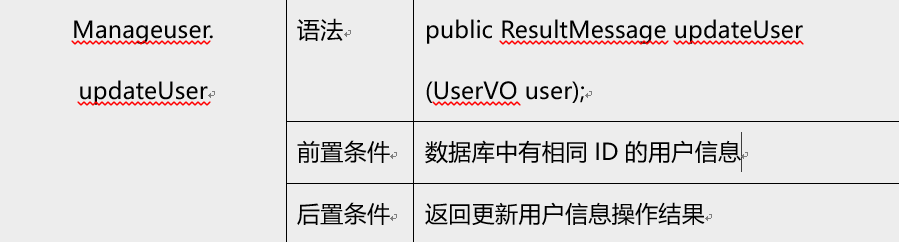
PreOrder方法的返回类型要改成OrderPO。线下入住也会调用preOrder方法但传入的OrderVO里面OrderStatus是OFFLINE，之后跟线上的一样，分配房间保存订单信息什么的，最后要生成OrderPO返回给逻辑层。

Strategy数据库第一行是会员升级的信用值门槛，第二行是各个会员等级的别名，第三行是各个会员等级对应的折扣比例。这三行在数据库初始化时就要默认添加。

## 12/5

出现rank1/rank2的方法需要返回第rank1个到第rank2个元素（包括第rank1/rank2个元素）

ManageUser增加updateUser方法，用于更新用户信息。



UserMain删除signUp方法，由ManageUser. addUser实现。

DistrictVO以二进制文件形式在客户端本地存储。

酒店ID包含四位城市序号，两位商圈序号，三位酒店序号；商圈ID包含四位城市序号，两位商圈序号。