## 二、中层设计之逻辑层设计

### 2.7 OfflineRecord模块的静态结构与动态行为

#### 2.7.1 OfflineRecordService

##### 2.7.1.1 局部模块的职责

（1）模块概述

OfflineRecordService模块承担的需求参见需求规格说明文档功能需求及相关的非功能需求。OfflineRecordService模块的职责及接口参见软件系统结构描述文档表5.3.2-8

（2）整体结构

OfflineRecordService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。OfflineRecordService模块的职责及接口参见软件体系结构描述文档。根据体系结构的设计，采用分层风格，将系统分为展示层，业务逻辑层，数据层。每一层之间为了灵活性，添加了接口，以实现针对接口编程，隔离数据传输的职责，降低层与层之间耦合，添加了OfflineRecordService, OfflineRecordDAO两个接口。OfflineRecordPO是做为管理信息的持久化对象被添加到设计模型中的。

图2.7.1.1-1 OfflineRecordService模块设计

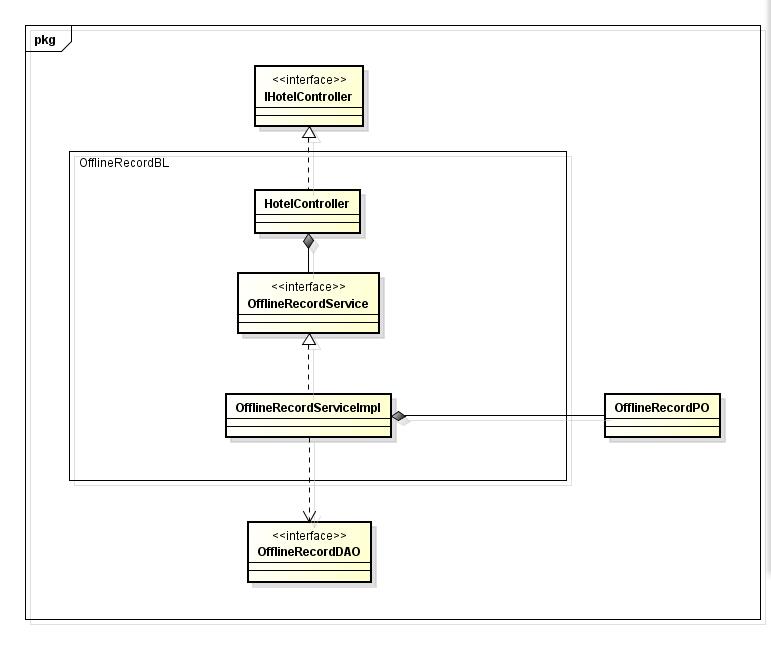


表2.3.1.1-2 OfflineRecordService模块各个类的职责

|  |  |
| --- | --- |
| 类 | 职责 |
| HotelController | 负责实现对应酒店线下记录界面所需的方法 |
| OfflineRecordServiceImpl | 负责处理有关线下记录的功能 |

##### 2.7.1.2 局部模块的接口规范

表2.7.1.2 -1 OfflineRecordService的接口规范

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| OfflineRecordService. findByID | 语法 | public OfflineRecordVO findByID(int id) | |
| 前置条件 | 已知线下记录编号，需要线下记录的信息 | |
| 后置条件 | 返回该编号对应的线下记录信息 | |
| OfflineRecordService. add | 语法 | public ResultMessage add(OfflineRecordVO) | |
| 前置条件 | 已知线下记录信息，需要将其添加到数据库 | |
| 后置条件 | 返回添加的结果信息 | |
| OfflineRecordService.checkin | 语法 | public ResultMessage checkin(OfflineRecordVO) | |
| 前置条件 | 已知线下记录信息，需要将该记录状态改为已入住 | |
| 后置条件 | 在数据库中更新线下记录信息 | |
| OffLineRecordService.checkout | 语法 | public ResultMessage checkout(OfflineRecordVO) | |
| 前置条件 | 已知线下记录信息，需要将该记录状态改为已退房 | |
| 后置条件 | 在数据库中更新线下记录信息 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 |
| OfflineRecordDAO. findByID(int id) | | | 查找单一持久化序列 |
| OfflineRecordDAO.add(OfflineRecordPO) | | | 添加单一持久化序列 |
| OfflineRecordDAO.update(OfflineRecordPO) | | | 更新单一持久化序列 |

##### 2.7.1.3 局部模块的行为

图2.7.1.3-1 OfflineRecordService模块时序图1

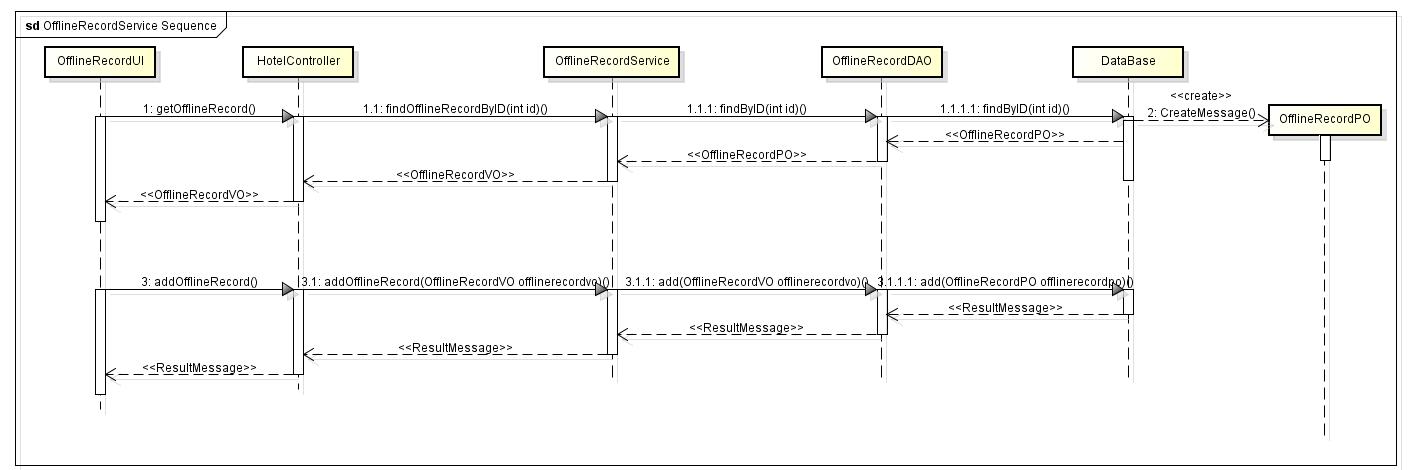


图2.7.1.3-1 表明了酒店预订系统中，当酒店工作人员查询和添加线下记录时，线下记录业务逻辑处理的相关对象之间的协作。

图2.7.1.3-2 OfflineRecordService模块时序图2

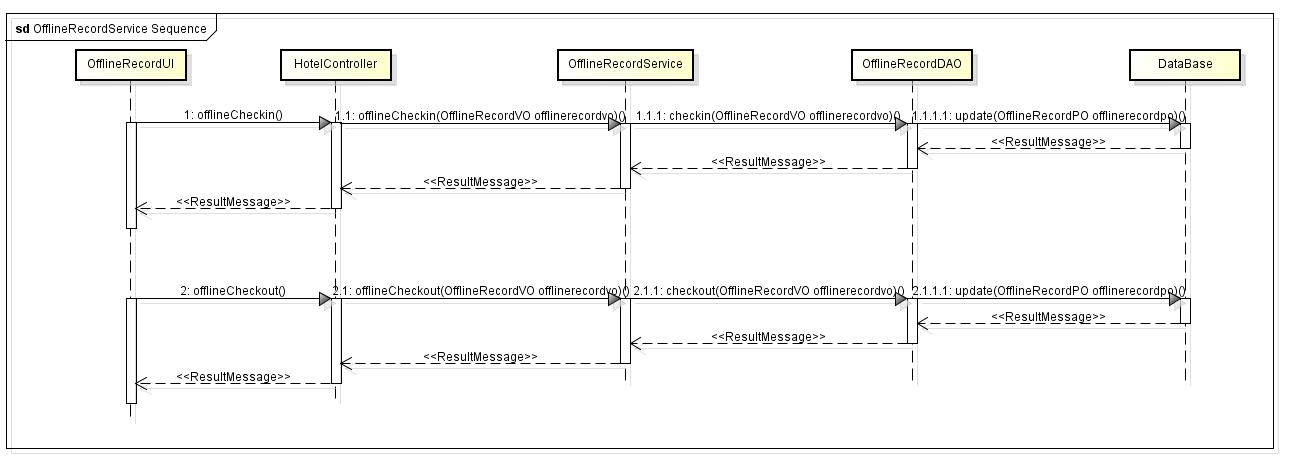


图2.7.1.3-2 表明了酒店预订系统中，当酒店工作人员修改线下记录时，线下记录业务逻辑处理的相关对象之间的协作。

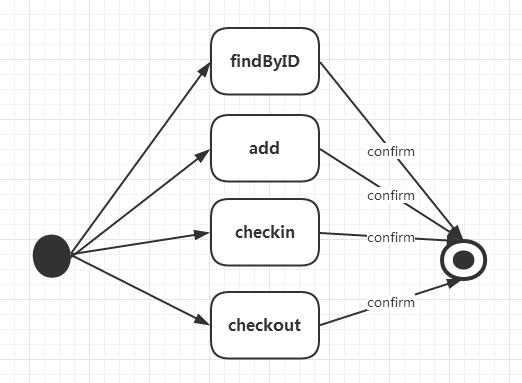


图2.7.1.3-3 OfflineRecordService模块状态图

### 2.8 Staff模块的静态结构与动态行为

#### 2.8.1 StaffService

##### 2.8.1.1 局部模块的职责

（1）模块概述

StaffService模块承担的需求参见需求规格说明文档功能需求及相关的非功能需求。StaffService模块的职责及接口参见软件系统结构描述文档表5.3.2-5

（2）整体结构

StaffService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。StaffService模块的职责及接口参见软件体系结构描述文档。根据体系结构的设计，采用分层风格，将系统分为展示层，业务逻辑层，数据层。每一层之间为了灵活性，添加了接口，以实现针对接口编程，隔离数据传输的职责，降低层与层之间耦合，添加了StaffService, StaffDAO两个接口。StaffPO是做为管理信息的持久化对象被添加到设计模型中的。

图2.8.1.1-1 StaffService模块设计

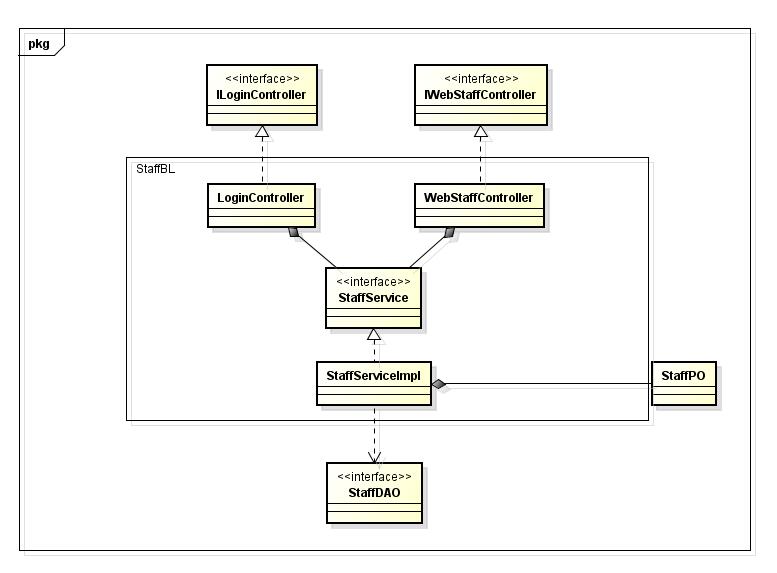


表2.8.1.1-2 StaffService模块各个类的职责

|  |  |
| --- | --- |
| 类 | 职责 |
| LoginController | 负责实现对应登录注册界面所需的方法 |
| WebStaffController | 负责实现对应酒店工作人员、网站营销人员信息管理界面 |
| StaffServiceImpl | 负责处理有关酒店工作人员、网站管理人员管理用户、网站营销人员的功能 |

##### 2.7.1.2 局部模块的接口规范

表2.8.1.2 -1 StaffService的接口规范

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| StaffService.login | 语法 | public StaffVO login(String username,String password) | |
| 前置条件 | 已知用户名和密码，需要进行登录 | |
| 后置条件 | 无 | |
| StaffService.update | 语法 | public ResultMessage update(StaffVO) | |
| 前置条件 | 已知更新后的职员信息，需要更新数据库中的信息 | |
| 后置条件 | 在数据库中更新职员信息 | |
| StaffService.add | 语法 | public ResultMessage add(StaffVO) | |
| 前置条件 | 已知职员信息，需要将其添加到数据库中 | |
| 后置条件 | 在数据库中保存新添加的职员信息 | |
| StaffService.findByUsername | 语法 | public StaffVO findByUsername(String username) | |
| 前置条件 | 已知用户名，需要返回对应的职员信息 | |
| 后置条件 | 无 | |
| StaffService.findByHotelName | 语法 | public StaffVO findByHotelName(String hotelName) | |
| 前置条件 | 已知酒店名称，需要返回对应的职员信息（如果是酒店工作人员） | |
| 后置条件 | 无 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 |
| StaffDAO.update(StaffPO) | | | 更新单一持久化序列 |
| StaffDAO. add(StaffPO) | | | 添加单一持久化序列 |
| StaffDAO. findByUsername(String username) | | | 查找单一持久化序列 |
| StaffDAO.findByHotelName(String hotelName); | | | 查找单一持久化序列 |

##### 2.8.1.3 局部模块的行为

图2.8.1.3-1 StaffService模块时序图1

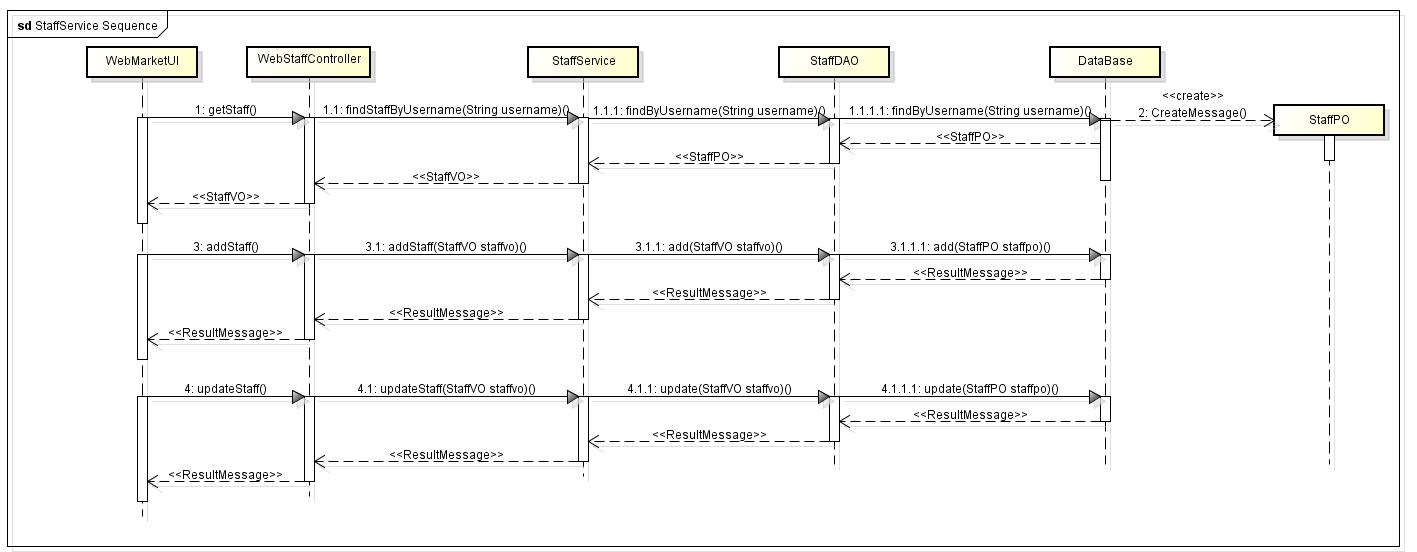


图2.8.1.3-1 表明了酒店预订系统中，当网站管理人员查询、修改和增加网站营销人员信息时，职员业务逻辑处理的相关对象之间的协作。

图2.8.1.3-2 StaffService模块时序图2

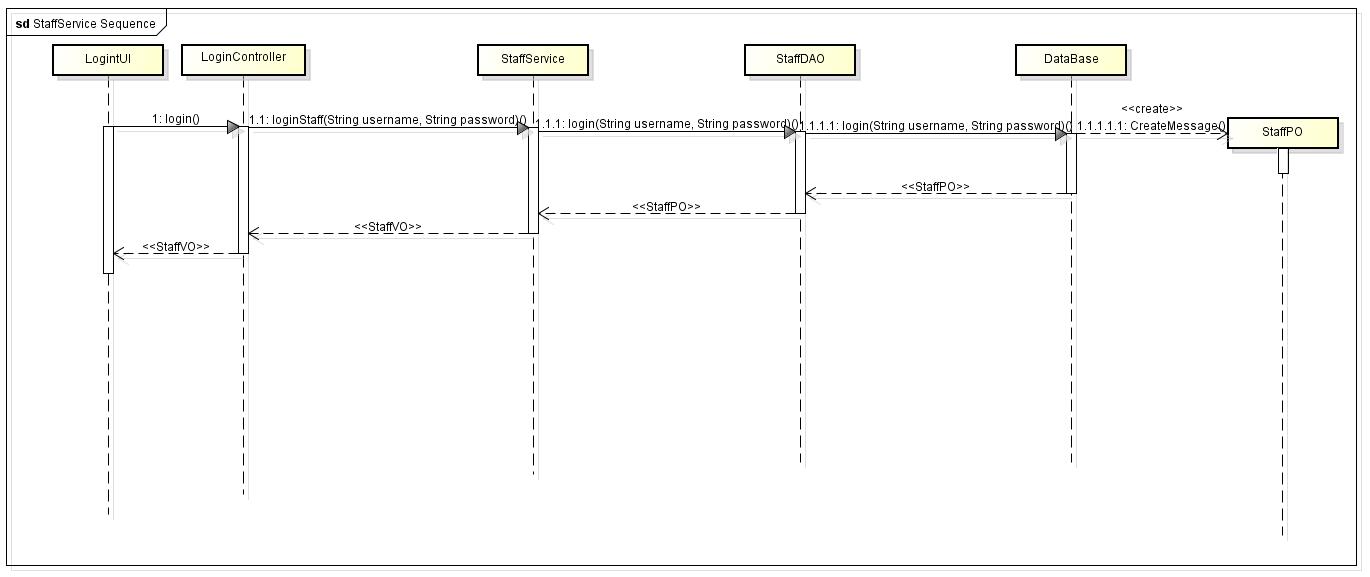


图2.8.1.3-2 表明了酒店预订系统中，当职员登录时，职员业务逻辑处理的相关对象之间的协作。

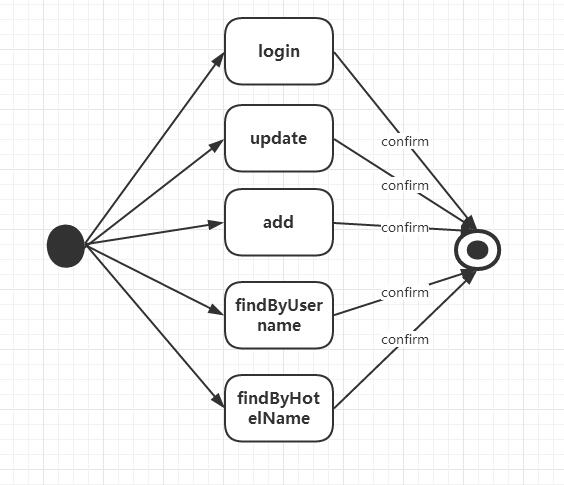


图2.8.1.3-3 StaffService模块状态图

## 三、中层设计之控制器设计

### 3.5 WebStaffUI包所对应的Controller

#### 3.5.1 WebUserController

图3.5.1-1 WebUserController 类图

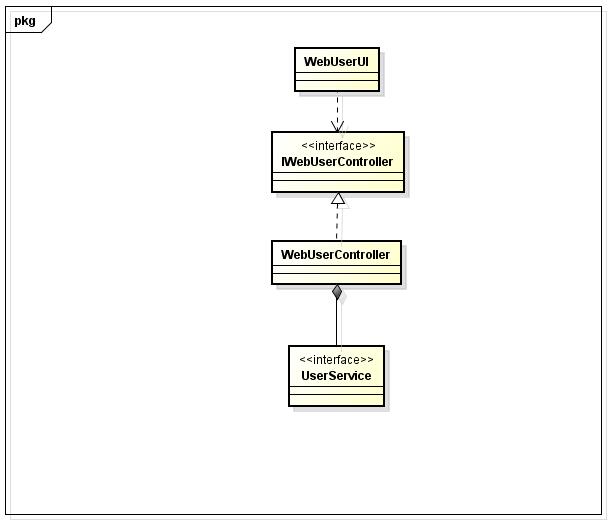


表3.5.1 -2 WebUserController的接口规范

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| WebUserController.findUserByUsername | 语法 | public UserVO findUserByUsername(String username) | |
| 前置条件 | 已知用户名，需要返回该用户的信息 | |
| 后置条件 | 无 | |
| WebUserController.updateUser | 语法 | public ResultMessage updateUser(UserVO uservo) | |
| 前置条件 | 已知更新后的信息，需要将其添加到数据库 | |
| 后置条件 | 在数据库中更新用户信息 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 |
| UserService.update(UserVO) | | | 更新单一持久化序列 |
| UserService.findByUsername(String username) | | | 查找单一持久化序列 |

#### 3.5.2 WebStaffController

图3.5.2-1 WebStaffController 类图

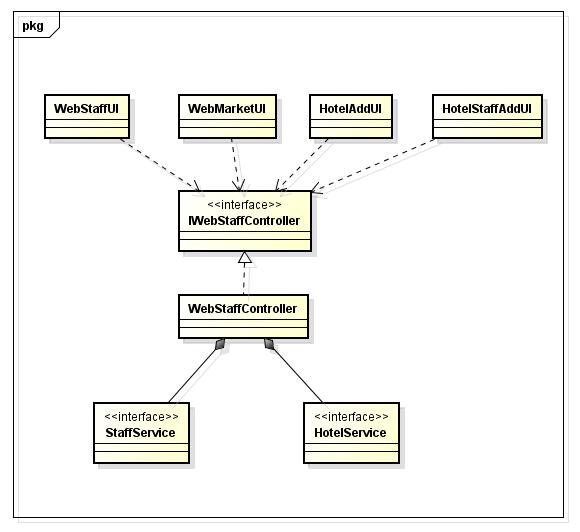


表3.5.2 -2 WebStaffController的接口规范

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| WebStaffController.addHotel | 语法 | Public ResultMessage addHotel(HotelVO hotelvo) | |
| 前置条件 | 已知酒店信息，需要将其添加到数据库 | |
| 后置条件 | 在数据库中更新酒店信息 | |
| WebStaffController.addStaff | 语法 | public ResultMessage addStaff(StaffVO staffvo) | |
| 前置条件 | 已知职员信息，需要将其添加到数据库 | |
| 后置条件 | 在数据库中更新职员信息 | |
| WebStaffController.updateStaff | 语法 | Public ResultMessage updateStaff(StaffVO staffvo) | |
| 前置条件 | 已知更新后的信息，需要将其添加到数据库 | |
| 后置条件 | 在数据库中更新职员信息 | |
| WebStaffController.findStaffByUsername | 语法 | Public StaffVO findStaffByUsername(String username) | |
| 前置条件 | 已知职员的用户名，需要返回该职员的信息 | |
| 后置条件 | 无 | |
| WebStaffController.findStaffByHotelName | 语法 | Public StaffVO findStaffByHotelName(String hetelName) | |
| 前置条件 | 已知酒店工作人员的酒店名称，需要返回该酒店工作人员的信息 | |
| 后置条件 | 无 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 |
| HotelService.add(HotelVO hotelvo) | | | 添加单一持久化序列 |
| StaffService.add(StaffVO staffvo) | | | 添加单一持久化序列 |
| StaffService.update(StaffVO staffvo) | | | 更新单一持久化序列 |
| StaffService.findByUsername(String username) | | | 查找单一持久化序列 |
| StaffService.findByHotelName(String hotelName) | | | 查找单一持久化序列 |