### 5.4 数据层的分解

数据层主要给业务逻辑层提供数据防伪服务，包括对于持久化数据的增、删、改、查。User业务逻辑需要的服务由UserDataService接口提供。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。数据层模块的具体描述如图5.4所示。



图5.4 数据层模块的描述

#### 5.4.1 数据层模块的职责

数据层模块的职责如表5.4.1所示。

表5.4.1 数据层模块的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| DataService | 持久化数据库的接口，提供集体载入，集体保存、增、删、改、查服务。 |
| DataServiceTxtImpl | 基于TXT文件的持久化数据库的接口，提供集体载入、集体保存、增、删、改、查服务 |
| DataServiceSpringImpl | 基于数据库的持久化数据的接口，提供集体载入、集体保存、增、删、改、查服务 |

#### 5.4.2 数据层模块的接口规范

**5.4.2 数据层模块的接口规范**

表5.4.2-1数据层UserDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| UserDAO.findByUsername | 语法 | Public UserPO findByUsername  (String username) |
| 前置条件 | 无 |
| 后置条件 | 返回对应username的PO记录 |
| UserDAO.add | 语法 | public ResultMessage add(UserPO) |
| 前置条件 | 逻辑层请求添加新的user信息，且同样的username的PO在数据文件中不存在 |
| 后置条件 | 在数据文件中增加一个user的PO记录 |
| UserDAO.update | 语法 | Public ResultMessage update(UserPO) |
| 前置条件 | 逻辑层请求更新user的信息，且对应的userPO在数据文件中已存在 |
| 后置条件 | 在数据文件中更新该user的PO记录 |

表5.4.2-2数据层OrderDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| OrderDAO.findByID | 语法 | public OrderPO findByID(int ID); |
| 前置条件 | 无 |
| 后置条件 | 返回对应ID的OrderPO,如果ID不存在，返回Null |
| OrderDAO.findByUsernameAndType | 语法 | Public List<OrderPO> findByUsernameAndType(String username,OrderStatus) |
| 前置条件 | 无 |
| 后置条件 | 返回与username和type对应的OrderPO记录 |
| OrderDAO.findByHotelAndUsername | 语法 | public List<OrderPO> findByHotelAndUsername(int hotelID,String username) |
| 前置条件 | 无 |
| 后置条件 | 返回与hotelID和username相对应的OrderPO记录 |
| OrderDAO.findByUsername | 语法 | Public List<OrderPO> findByUsername(String username) |
| 前置条件 | 无 |
| 后置条件 | 返回与username对应的OrderPO记录 |
| OrderDAO.findByOrderType | 语法 | public List<OrderPO>  findByOrderType(OrderStatus) |
| 前置条件 | 无 |
| 后置条件 | 返回与OrderStatus对应的PO记录 |
| OrderDAO.findByHotelAndTime | 语法 | public List<OrderPO> findByHotelAndTime(int hotelID,Date begin,Date end); |
| 前置条件 | 无 |
| 后置条件 | 返回与hotelID和time对应的PO记录 |
| OrderDAO.add | 语法 | public ResultMessage add  (OrderPO); |
| 前置条件 | 逻辑层请求添加订单记录，且同样ID的PO在数据文件中不存在 |
| 后置条件 | 在数据库中添加一个Order记录 |
| OrderDAO.update | 语法 | public ResultMessage update  (OrderPO) |
| 前置条件 | 逻辑层请求更新订单记录，且对应的  OrderPO记录在数据文件中已存在 |
| 后置条件 | 在数据文件中更新OrderPO记录 |

表5.4.2-3数据层HotelDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| HotelDAO.findByID | 语法 | public HotelPO findByID(int hotelID); |
| 前置条件 | 无 |
| 后置条件 | 返回与hotelID对应的HotelPO记录 |
| HotelDAO.update | 语法 | public ResultMessage update  (HotelPO); |
| 前置条件 | 逻辑层请求更新hotelPO记录，且对应hotelPO记录在数据文件中已存在 |
| 后置条件 | 在数据文件中更新对应hotel的PO记录 |
| HotelDAO.add | 语法 | public ResultMessage add  (HotelPO); |
| 前置条件 | 逻辑层请求添加新的hotel记录，且相同ID的hotel在数据文件中不存在 |
| 后置条件 | 在数据文件中增加一个PO记录 |
| HotelDAO.find | 语法 | public List<HotelPO> find (int loc,int circle); |
| 前置条件 | 无 |
| 后置条件 | 返回与loc，circle对应的hotelPO记录 |

表5.4.2-4数据层LocationDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| LocationDAO.findAll | 语法 | public List<LocationPO> findAll(); |
| 前置条件 | 无 |
| 后置条件 | 返回所有Location的PO记录 |

表5.4.2-5数据层CommCircleDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| CommCircleDAO. findByLoc | 语法 | public List<CommercialCirclePO>  findByLoc(int locID); |
| 前置条件 | 无 |
| 后置条件 | 返回对应locID的CommercialCirclePO记录 |

表5.4.2-6数据层RoomDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| RoomDAO.findByHotelAndType | 语法 | public RoomPO find(int hotelID,  RoomType); |
| 前置条件 | 无 |
| 后置条件 | 返回与hotelID和RoomType对应的  RoomPO记录 |
| RoomDAO.update | 语法 | public ResultMessage update  (RoomPO); |
| 前置条件 | 逻辑层请求更新Room的PO记录，且对应ID的PO在数据文件中已存在 |
| 后置条件 | 在数据文件中更新PO |
| RoomDAO.add | 语法 | public ResultMessage add  (RoomPO); |
| 前置条件 | 逻辑层请求添加新的room记录，且相同ID的PO在数据文件中不存在 |
| 后置条件 | 在数据文件中增加一个PO记录 |
| RoomDAO.findByHotelID | 语法 | Public LIst<RoomPO>  findByHotelID(int hotelID) |
| 前置条件 | 无 |
| 后置条件 | 返回对应hotelID的RoomPO记录 |

表5.4.2-7数据层StaffDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| StaffDAO.update | 语法 | public ResultMessage update(StaffPO); |
| 前置条件 | 逻辑层请求更新Staff的PO记录，且对应ID的PO在数据文件中已存在 |
| 后置条件 | 在数据文件中更新PO |
| StaffDAO.add | 语法 | public ResultMessage add(StaffPO); |
| 前置条件 | 逻辑层请求添加新的staff记录，且相同ID的PO在数据文件中不存在 |
| 后置条件 | 在数据文件中增加一个PO记录 |
| StaffDAO.findByUsername | 语法 | public StaffPO findByUsername(String username); |
| 前置条件 | 无 |
| 后置条件 | 返回与username对应的StaffPO记录 |

表5.4.2-8数据层CreditRecordDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| CreditRecordDAO.  findByUsername | 语法 | public List<CreditRecordPO> findByUsername(String username); |
| 前置条件 | 无 |
| 后置条件 | 返回与username对应CreditRecord  PO记录 |
| CreditRecordDAO.add | 语法 | public ResultMessage add  (CreditRecordPO); |
| 前置条件 | 逻辑层请求增加新的信用记录 |
| 后置条件 | 在数据文件中增加一条信用记录 |

表5.4.2-9数据层HotelDiscountDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| HotelDiscountDAO.add | 语法 | public ResultMessage add  (HotelDiscountPO); |
| 前置条件 | 逻辑层请求添加新的HotelDiscount记录，且相同ID的PO在数据文件中不存在 |
| 后置条件 | 在数据文件中增加一个PO记录 |
| HotelDiscountDAO.update | 语法 | public ResultMessage update  (HotelDiscountPO) |
| 前置条件 | 逻辑层请求更新HotelDiscount的PO记录，且对应ID的PO在数据文件中已存在 |
| 后置条件 | 在数据文件中更新PO |
| HotelDiscountDAO.delete | 语法 | public ResultMessage delete(int id); |
| 前置条件 | 相同ID的PO在数据文件中已存在 |
| 后置条件 | 删除一个PO记录 |
| HotelDiscountDAO.findAllByHotelID | 语法 | public List<HotelDiscountPO>  findAllByHotelID(int hotelID); |
| 前置条件 | 无 |
| 后置条件 | 返回对应hotelID的HotelDiscount  PO记录 |

表5.4.2-10数据层EnterpriseDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| EnterpriseDAO. findAll | 语法 | public List< EnterprisePO > findAll(); |
| 前置条件 | 无 |
| 后置条件 | 返回所有EnterprisePO记录 |

表5.4.2-11数据层WebDiscountDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| WebDiscountDAO.add | 语法 | public ResultMessage add  (WebDiscountPO); |
| 前置条件 | 逻辑层请求添加新的WebDiscount记录，且同样的WebDiscountPO在数据文件中不存在 |
| 后置条件 | 在数据文件中增加一个PO记录 |
| WebDiscountDAO  .update | 语法 | public ResultMessage update  (WebDiscountPO); |
| 前置条件 | 逻辑层请求更新WebDiscount,且对应ID的PO记录在数据文件中已存在 |
| 后置条件 | 在数据文件中更新 WebDiscou的PO记录 |
| WebDiscountDAO  .delete | 语法 | public ResultMessage delete(int id); |
| 前置条件 | 相同ID的PO在数据文件中已存在 |
| 后置条件 | 删除一个PO记录 |
| WebDiscountDAO  .findAll | 语法 | public List<WebDiscountPO>  findAll(); |
| 前置条件 | 无 |
| 后置条件 | 返回所有WebDiscountPO记录 |

表5.4.2-12数据层OfflineRecordDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| OfflineRecordDAO.findByID | 语法 | public OfflineRecordPO findByID(int id); |
| 前置条件 | 无 |
| 后置条件 | 返回与ID对应的线下记录PO |
| OfflineRecordDAO.add | 语法 | public ResultMessage add  (OfflineRecordPO); |
| 前置条件 | 逻辑层请求添加新的OfflineRecord记录，且同样的线下记录PO在数据文件中不存在 |
| 后置条件 | 在数据文件中增加一个PO记录 |
| OfflineRecordDAO.update | 语法 | Public ResultMessage update(OfflineRecordPO); |
| 前置条件 | 逻辑层请求更新线下记录，且对应的PO在数据文件中已存在 |
| 后置条件 | 数据文件中更新线下记录PO |

表5.4.2-13数据层AvailableRoomDAO模块的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| AvailableRoomDAO.findAvailableRoom | 语法 | Public int findAvailableRoom(int hotelID,RoomType,Date date); |
| 前置条件 | 无 |
| 后置条件 | 返回与hotelID，RoomType,Date对应的所有可用房间PO记录 |