4.2.1 accountData模块

（1）模块概述

accountData模块承担的需求参见需求规格说明文档需求及相关非功能需求。

accountData模块的职责及接口参见软件体系结构描述文档5.3.2中accountData模块的接口规范。

（2）整体结构

根据体系结构的设计，分层结构分为界面层，逻辑层和数据层。界面层和逻辑层之间添加accountLogicSer作为账户管理界面的逻辑层接口。数据层和逻辑层之间添加accountDSer作为账户逻辑的数据层接口。AccountDataSerImpl是基于Mysql数据库的持久化数据接口，实现AccountDataSer的服务。accountPO作为用户的持久化对象。

accountData模块设计如图所示。



accountData模块各个类的职责如表所示。

accountData各个类的职责

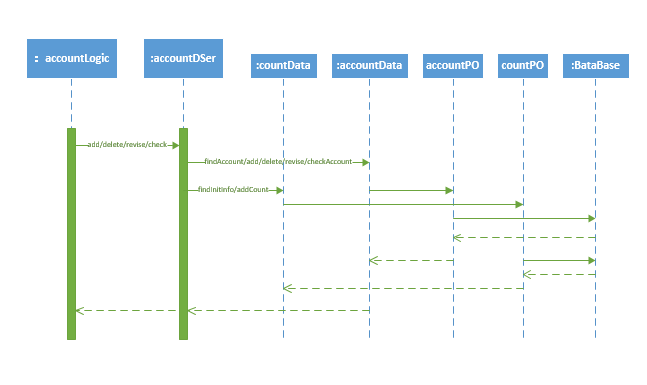
|  |  |
| --- | --- |
| 模块 | 职责 |
| accountData | 负责账户管理数据处理 |
| countData | 负责实现账单管理的数据处理 |

AccountDataSer的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| AccountDataSer.findAccount | 语法 | public ArrayList<AccountPO> findAccount(String keyword) throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 按关键字查找返回符合条件的账户列表 |
| AccountDataSer.addAccount | 语法 | public boolean addAccount(AccountPO account) throws RemoteException |
| 前置条件 | 同样账户名的账户在数据库中不存在 |
| 后置条件 | 在数据库中增加一个po记录 |
| AccountDataSer.deleteAccount | 语法 | public boolean deleteAccount(String id) throws RemoteException |
| 前置条件 | 该id对应的账户在数据库中存在 |
| 后置条件 | 在数据库中删除一个po记录 |
| AccountDataSer.reviseAccount | 语法 | public boolean reviseAccount(AccountPO po) throws RemoteException |
| 前置条件 | 该账户在数据库中存在 |
| 后置条件 | 在数据库中更新该账户信息 |
| AccountDataSer.checkAccount | 语法 | public AccountPO checkAccount(String id) throws RemoteException |
| 前置条件 | 该id对应的账户在数据库中存在 |
| 后置条件 | 返回该id对应的账户信息 |
| AccountDataSer.findInitInfo | 语法 | public AcountPO findInitInfo() throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 返回该id对应的期初信息 |
| AccountDataSer.addCount | 语法 | public boolean addCount(CountPO count) throws RemoteException |
| 前置条件 | 同样账户名的账户在数据库中不存在 |
| 后置条件 | 在数据库中增加一个po记录 |
| AccountDataSer.findOrg | 语法 | public ArrayList<OrgPO> findOrg() throws RemoteException |
| 前置条件 |  |
| 后置条件 | 在数据库中获取并返回机构信息 |
| AccountDataSer.getWorkers | 语法 | Public ArrayList<WorkPO> getWorkers(String id)throws RemoteException |
| 前置条件 | 账户名的账户在数据库中存在 |
| 后置条件 | 在数据库中获取并返回人员信息 |
| AccountDataSer.getVans | 语法 | public ArrayList<VanPO> getVans(String id) throws RemoteException |
| 前置条件 |  |
| 后置条件 | 在数据库中获取并返回货车信息 |
| AccountDataSer.getStock | 语法 | Public ArrayList<StockPO> getStock(String id)throws RemoteException |
| 前置条件 |  |
| 后置条件 | 在数据库中获取并返回库存信息 |
| AccountDataSer.getAccount | 语法 | public ArrayList<AccountPO> getAccount(String id) throws RemoteException |
| 前置条件 |  |
| 后置条件 | 在数据库中获取并返回账户信息 |

（4）数据层的动态模型

账户管理顺序图，表明了账户管理过程相关数据对象之间的协作。



4.2.2 orderApproveData模块

（1）模块概述

orderApproveData模块承担的需求参见需求规格说明文档需求及相关非功能需求。

orderApproveData模块的职责及接口参见软件体系结构描述文档5.3.2中orderApproveData模块的接口规范。

（2）整体结构

根据体系结构的设计，分层结构分为界面层，逻辑层和数据层。界面层和逻辑层之间添加orderApproveLogicSer作为账户管理界面的逻辑层接口。数据层和逻辑层之间添加orderApproveDSer作为账户管理逻辑的数据层接口。orderApproveDataSerImpl是基于Mysql数据库的持久化数据接口，实现OrderApproveDataSer的服务。orderPO作为用户的持久化对象。

orderApproveData模块各个类的职责如表所示。

orderApproveData各个类的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| OrderApproveData | 负责单据审批数据处理 |
| OrderResubmitData | 负责实现单据重新提交的数据处理 |

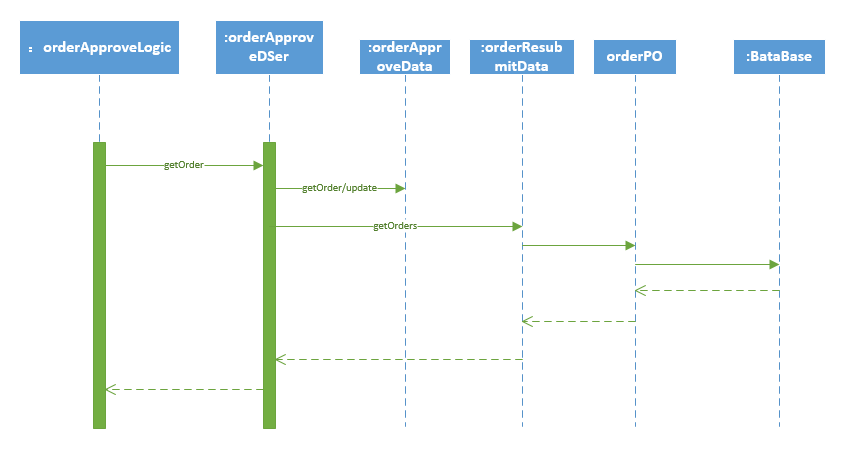
orderApproveData模块设计如图所示。

OrderApproveDataSer模块接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| OrderApproveDataSer.update | 语法 | public boolean update(ArrayList<OrderPO> po)throws RemoteException |
| 前置条件 | Order存在于数据库中 |
| 后置条件 | 在数据库中更新订单审批状态 |
| OrderApproveDataSer.getOrder | 语法 | public OrderPO getOrder(String id) throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 从数据库获得并返回相应的单据信息 |
| OrderApproveDataSer.getOrders | 语法 | public ArrayList<OrderPO> getOrder(String id) throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 从数据库获得并返回相应的单据信息 |
| OrderApproveDataSer.getExamine | 语法 | public ArrayList<OrderPO> getExamine (String id) throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 从数据库获得并返回相应的单据审批信息 |

（4）数据层的动态模型

单据审批顺序图，表明了单据审批过程相关数据对象之间的协作。



4.2.3 transitInfoData模块

（1）模块概述

transitInfoData模块承担的需求参见需求规格说明文档需求及相关非功能需求。

transitInfoData模块的职责及接口参见软件体系结构描述文档5.3.2中transitInfoData模块的接口规范。

（2）整体结构

根据体系结构的设计，分层结构分为界面层，逻辑层和数据层。界面层和逻辑层之间添加transitInfoLogicSer作为账户管理界面的逻辑层接口。数据层和逻辑层之间添加transitInfoDSer作为账户管理逻辑的数据层接口。transitInfoDataSerImpl是基于Mysql数据库的持久化数据接口，实现TransitInfoDataSer的服务。

模块各个类的职责如表所示。

transitInfoData各个类的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| CourierData | 负责实现快递员的物流信息查询数据处理 |
| CustomerData | 负责实现用户的物流信息查询数据处理 |

transitInfoData模块设计如图所示。



TransitInfoDataSer的接口规范

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| TransitInfoDataSer.find | 语法 | public orderPO find(String barcode) throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 按barcode进行查找返回相应的orderPO结果 |
| TransitInfoDataSer.findTransit | 语法 | public ArrayList<String> findTransit(String barcode) throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 按barcode进行查找返回相应订单的物流轨迹和货运信息 |

（4）数据层的动态模型

物流信息查询顺序图，表明了物流信息查询过程相关数据对象之间的协作。

