4.1.3 Financialbl模块

（1）模块概述

Financialbl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。

Financialbl模块的职责及借款参见软件体系结构描述文档。

（2）整体结构

根据体系结构的设计，我们将系统分为展示层、业务逻辑层、数据层。每一层之间为了增加灵活性，我们会添加接口。Financial模块对应的接口为展示层与业务逻辑层之间的businesslogicservice.financialblservice.FinancialblService接口与业务逻辑层与数据层之间的dataservice.financialdataservice.FinancialblService接口。为了隔离业务逻辑职责和逻辑控制职责，我们增加了FinancialController，这样FinancialController会将对账户的业务逻辑处理委托给Financial对象。SalesListPO是作为销售明细的持久化对象被添加到设计模型中去的。AllBillPO是作为经营历程的持久化对象被添加到设计模型中去的。OperatingConditionPO是作为经营情况的持久化对象被添加到设计模型中去的。SystemlogInfo、AccountInfo、SalesInfo、CommodityInfo都是根据依赖倒置原则，为了消除循环依赖而产生的接口。

Financialbl模块的设计如图

Financialbl模块各个类的职责如表

|  |  |
| --- | --- |
| 模块 | 职责 |
| FinancialController | 负责实现财务表单界面所需要的服务 |
| Financial | 财务账单领域模型对象，拥有经过加工后的账单信息，可以帮助完成财务表单界面所需要的服务 |

（3）模块内部类的接口规范

FinancialController接口规范：

|  |  |  |
| --- | --- | --- |
| 提供的服务 | | |
| FinancialController.saleList | 语法 | Public ResultMessage saleList(String time1, String time2, String good\_name, String good\_type, String customer\_name, String clerk, int warehouse) |
| 前置条件 | 已创建一个Financial领域对象，输入符合规则 |
| 后置条件 | 调用Financial领域对象的saleList方法 |
| FinancialController.allBill | 语法 | Public ResultMessage allBill(String time1, String time2, String note\_type, String customer\_name, String clerk, int warehouse) |
| 前置条件 | 已创建一个Financial领域对象，输入符合规则 |
| 后置条件 | 调用Financial领域对象的allBill方法 |
| FinancialController.operatingCondition | 语法 | Public ResultMessage operatingCondition(String time1, String time2) |
| 前置条件 | 已创建一个Financial领域对象，输入符合规则 |
| 后置条件 | 调用Financial领域对象的operatingCondition方法 |
| 需要的服务 | | |
| Financial.saleList(String time1, String time2, String good\_name, String good\_type, String customer\_name, String clerk, int warehouse) | 根据条件查询销售明细 | |
| Financial.allBill(String time1, String time2, String note\_type, String customer\_name, String clerk, int warehouse) | 根据条件查询经营历程 | |
| Financial.operatingCondition(String time1, String time2) | 根据条件查询经营情况 | |

Financial接口规范：

|  |  |  |
| --- | --- | --- |
| 提供的服务 | | |
| Financial.saleList | 语法 | Public ResultMessage saleList(String time1, String time2, String good\_name, String good\_type, String customer\_name, String clerk, int warehouse) |
| 前置条件 | 操作员是经过验证的财务人员 |
| 后置条件 | 查询销售明细，返回操作是否成功 |
| Financial.allBill | 语法 | Public ResultMessage allBill(String time1, String time2, String note\_type, String customer\_name, String clerk, int warehouse) |
| 前置条件 | 操作员是经过验证的财务人员 |
| 后置条件 | 查询单据，返回操作是否成功 |
| Financial.operatingCondition | 语法 | Public ResultMessage operatingCondition(String time1, String time2) |
| 前置条件 | 操作员是经过验证的财务人员 |
| 后置条件 | 查看经营情况，返回操作是否成功 |
| 需要的服务 | | |
| FinancialDataService.getSaleList() | 获得所有销售明细对象 | |
| FinancialDataService.getAllBill() | 获得所有经营历程对象 | |
| FinancialDataService.getOperatingCondition() | 获得所有经营情况对象 | |
| FinancialDataService.addSaleList(SaleListPO po) | 添加销售明细的单据对象 | |
| FinancialDataService.addAllBill(AllBillPO po) | 添加经营历程的单据对象 | |
| FinancialDataService.addOperatingCondition(OperatingConditionPO po) | 添加经营情况的单据对象 | |
| FinancialDataService.clear() | 清空所有对象 | |