4.1.1orderbl模块

（1）模块概述

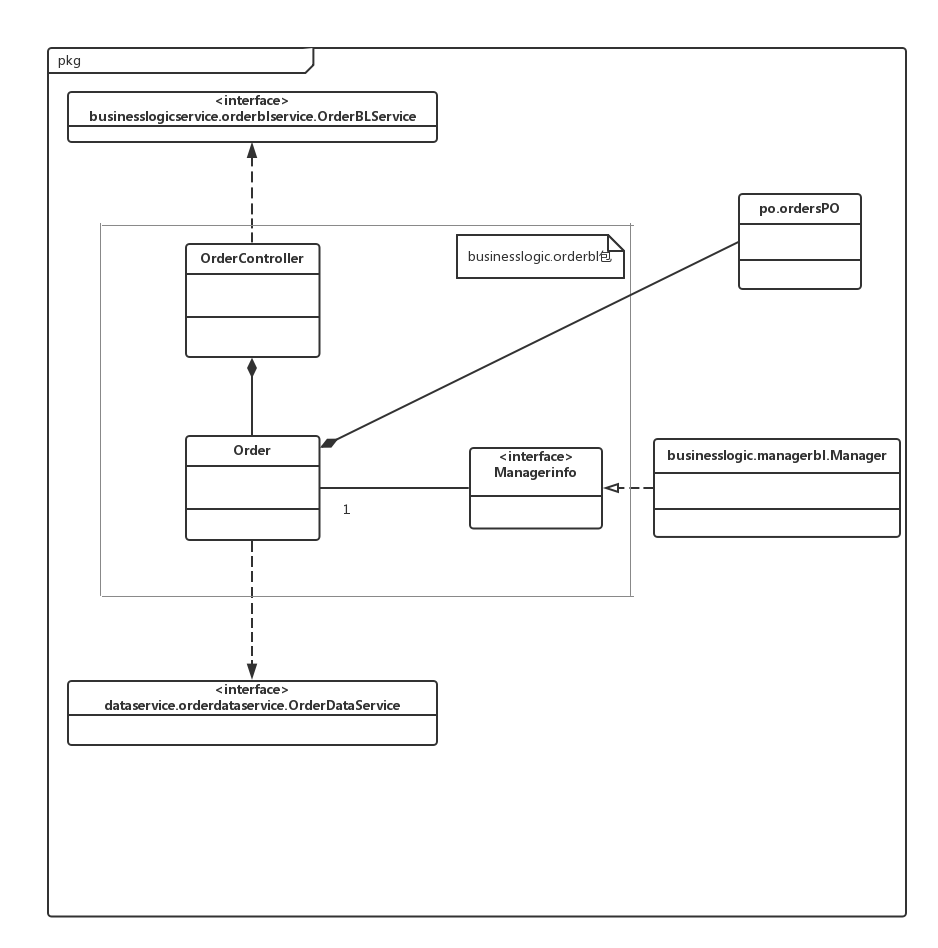
Orderbl模块承担的需求参加需求规格说明文档功能需求及其相关非功能需求。

Orderbl模块的职责及接口参见软件系统结构描述文档。

（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。比如展示曾和业务逻辑层之间，我们添加businesslogicservice.orderblservice.OrderBLService接口。业务逻辑层和数据层之间添加dataservice.orderdataservice.OrderDataService。为了隔离业务逻辑职责和逻辑控制职责，我们增加了OrderController，这样OrderController会将对销售的业务逻辑处理委托给Order对象。ordersPO是作为订单记录的持久化对象被添加到设计模型去的。Managerinfo是根据以来倒置原则，为了消除循环以来而产生的接口。

orderbl模块的设计如下图所示



orderbl模块各个类的设计

orderbl模块各个类的职责如下表所示。

**orderbl模块各个类的职责**

|  |  |
| --- | --- |
| **模块** | **职责** |
| LoginController | 负责实现对应于登录界所需要的服务 |
| OrderController | 负责实现订单输入界面所需要的服务 |
| User | 系统用户的领域模型对象，拥有用户数据的姓名和密码，可以解决登录问题 |
| Order | 订单输入的领域模型对象，拥有一次订单输入所持有的寄件人、收件人地址，快件重量，题记等信息，可以帮助完成订单输入界面所需要的服务 |

（3）模块内部类的接口规范

OrderController和Order的接口规范如下表所示。

**OrderController的接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| OrderController.order | 语法 | public ResultMessage order(OrderVO vo) | |
| 前置条件 | 已创建一个Order领域对象，输入信息符合规范 | |
| 后置条件 | 调用order领域对象的order方法 | |
| OrderController.. Computefare | 语法 | public double order(OrderFareVO vo) | |
| 前置条件 | 已创建一个Order领域对象，已添加收件人地址、快件重量、件数、体积、尺寸、包装类型、快递类型，输入信息符合规范 | |
| 后置条件 | 调用order领域对象的Computefare方法 | |
| OrderController.reception | 语法 | public ResultMessage reception (ReceptionVO vo) | |
| 前置条件 | 已创建一个Order领域对象，输入信息符合规范 | |
| 后置条件 | 调用order领域对象的reception方法 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 |
| Order.order(OrderVO vo) | | | 加入一个订单对象 |
| Order.Computefare(OrderFareVO vo) | | | 返回订单的运费 |
| Order.reception(ReceptionVO vo) | | | 更新订单对象的收件信息 |

**Order的接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| Order.order | 语法 | public ResultMessage order(OrderVO vo) | |
| 前置条件 | 已结束订单输入 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Order. Computefare | 语法 | public double order(OrderFareVO vo) | |
| 前置条件 | 已添加收件人地址、快件重量、件数、体积、尺寸、包装类型、快递类型 | |
| 后置条件 | 在一次订单输入中，返回此订单的运费 | |
| Order.reception | 语法 | public ResultMessage reception (ReceptionVO vo) | |
| 前置条件 | 已结束收件处理 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| 需要的服务（需接口） | | | |
| DatabaseFactory.getOrdersDa  Tabase | | | 得到Orders数据库的服务和引用 |
| OrdersDataService.addOrderForm(OrdersPO Po) | | | 向数据库中插入OrdersPO对象 |
| OrdersDataService.getOrderForm(Formstate state) | | | 从数据库中得到OrdersPO对象 |
| OrdersDataService.getOrderForm(long Ordernum) | | | 从数据库中得到OrdersPO对象 |
| OrdersDataService.updateOrderForm(OrderPO po) | | | 向数据库中更新OrdersPO对象 |

4.1.2managerbl模块

（1）模块概述

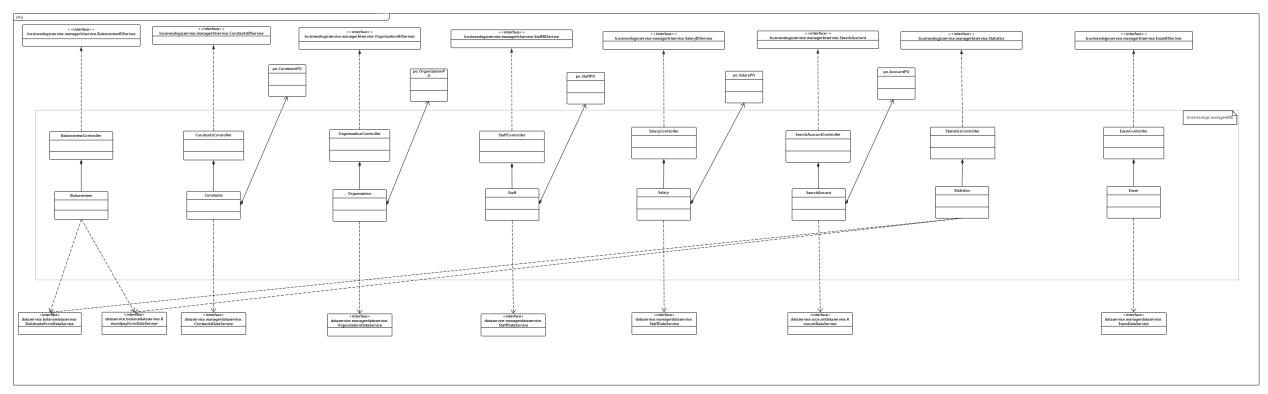
Managerbl模块承担的需求参加需求规格说明文档功能需求及其相关非功能需求。

Managerbl模块的职责及接口参见软件系统结构描述文档。

（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。比如展示曾和业务逻辑层之间，我们添加businesslogicservice.managerblservice.ManagerBLService接口。业务逻辑层和数据层之间添加dataservice.managerdataservice.ManagerDataService。为了隔离业务逻辑职责和逻辑控制职责，我们增加了BalanceviewController,ConstantsController,OrganizationController,StaffController,SalaryController,SearchAccountController,StatisticsController,ExamController，这样BalanceviewController,ConstantsController,OrganizationController,StaffController,SalaryController,SearchAccountController,StatisticsController,ExamController会将对销售的业务逻辑处理委托给Balanceview ,Constants ,Organization ,Staff ,Salary ,SearchAccount ,Statistics ,Exam 对象。ConstantsPO,OrganizationPO,StaffPO,SalaryPO,AccountPO是作为管理记录的持久化对象呗添加到设计模型去的。

Managerbl模块的设计如下图所示。



managerbl模块各个类的设计

Managerbl模块各个类的职责如下表所示。

**Managerbl模块各个类的职责**

|  |  |
| --- | --- |
| **模块** | **职责** |
| LoginController | 负责实现对应于登录界所需要的服务 |
| BalanceviewController | 负责实现查看成本收益表界面所需要的服务 |
| ConstantsController | 负责实现常量信息制定界面所需要的服务 |
| OrganizationController | 负责实现机构管理界面所需要的服务 |
| StaffController | 负责实现人员管理界面所需要的服务 |
| SalaryController | 负责实现薪水策略制定界面所需要的服务 |
| SearchAccoutController | 负责实现账户查询界面所需要的服务 |
| StatisticsController | 负责实现查看经营情况界面所需要的服务 |
| ExamController | 负责实现审批单据界面所需要的服务 |
| User | 系统用户的领域模型对象，拥有用户数据的姓名和密码，可以解决登录问题 |
| Balanceview | 查看成本收益表的领域模型对象，拥有一次查看成本收益表所需要的总收入、总支出、总利润等信息，可以帮助完成查看成本收益表界面所需要的服务 |
| Constants | 常量信息制定的领域模型对象，拥有一次查看常量信息制定所需要的常量名称、数值等信息，可以帮助完成常量信息制定界面所需要的服务 |
| Organization | 机构管理的领域模型对象，拥有一次机构管理所需要的机构名称、类型等信息，可以帮助完成机构管理界面所需要的服务 |
| Staff | 人员管理的领域模型对象，拥有一次人员管理所需要的人员名称、类型等信息，可以帮助完成人员管理界面所需要的服务 |
| Salary | 薪水策略制定的领域模型对象，拥有一次薪水策略制定所需要的薪水支付方式、职业等信息，可以帮助完成薪水策略制定界面所需要的服务 |
| SearchAccount | 账户查询的领域模型对象，拥有一次账户查询所需要的账户名称、密码等信息，可以帮助完成账户查询界面所需要的服务 |
| Statistics | 查看经营情况的领域模型对象，拥有一次经营情况查看所需要的付款单、收款单等信息，可以帮助完成查看经营情况界面所需要的服务 |
| Exam | 审批单据的领域模型对象，拥有一次审批单据所需要的单据状态、单据类型等信息，可以帮助完成审批单据界面所需要的服务 |

（3）模块内部类的接口规范

BalanceviewController和Balanceview的接口规范如下表所示。

**BalanceviewController的接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| BalanceviewController.getBalanceview | 语法 | public BalanceviewVO getBalanceview() | |
| 前置条件 | 已创建一个Balanceview领域对象 | |
| 后置条件 | 调用Balanceview领域对象的getBalanceview方法 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 | |
| Balanceview.getBalanceview() | | | 返回一个balanceviewVO对象 | |

**Balanceview的接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| Balanceview. getBalanceview | 语法 | public BalanceviewVO getBalanceview() | |
| 前置条件 | 启动一次成本收益表查看 | |
| 后置条件 | 返回一个balanceviewVO对象 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 | |
| balancedataservice.DebitnoteFormDataService. getDebitnoteForm | | | 返回所有的DebitnotePO | |
| balancedataservice.RecordpayFormDataService. getallRecordpayForm | | | 返回所有的RecordpayPO | |

ConstantsController和Constants的接口规范如下表所示。

**ConstantsController的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| ConstantsController.addConstants | 语法 | public ResultMessage addConstants (ConstantsVO vo) | | | |
| 前置条件 | 已创建一个Constants领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Constants领域对象的addContants方法 | | | |
| ConstantsController.delConstants | 语法 | public ResultMessage delConstants(ConstantsVO vo) | | | |
| 前置条件 | 已创建一个Constants领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Constants领域对象的delContants方法 | | | |
| ConstantsController.revConstants | 语法 | public ResultMessage revConstants (ConstantsVO vo) | | | |
| 前置条件 | 已创建一个Constants领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Constants领域对象的revContants方法 | | | |
| ConstantsController.getConstants | 语法 | public ConstantsVO getConstants (string name) | | | |
| 前置条件 | 已创建一个Constants领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Constants领域对象的getContants方法 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| Constants.addConstants | | | | 加入一个常量 | |
| Constants.delConstants | | | | 删除一个常量 | |
| Constants.revConstants | | | | 修改一个常量 | |
| Constants.getConstants | | | | 返回一个常量 | |

**Constants的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| Constants.addConstants | 语法 | public ResultMessage addConstants (ConstantsVO vo) | | | |
| 前置条件 | 完成新常量信息的输入 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Constants.delConstants | 语法 | public ResultMessage delConstants(ConstantsVO vo) | | | |
| 前置条件 | 确认删除一个常量信息 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Constants.revConstants | 语法 | public ResultMessage revConstants (ConstantsVO vo) | | | |
| 前置条件 | 完成常量信息的修改 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Constants.getConstants | 语法 | public ConstantsVO getConstants (string name) | | | |
| 前置条件 | 数据库中存在该常量 | | | |
| 后置条件 | 返回ConstantsVO对象 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| ManagerDataService.addConstants(ConstantsPO po) | | | | 向数据库中插入ConstantsPO对象 | |
| ManagerDataService.deleteConstants(ConstantsPO po) | | | | 从数据库中删除ConstantsPO对象 | |
| ManagerDataService.updateConstants(ConstantsPO po) | | | | 从数据库中修改更新某个ConstantsPO对象 | |
| ManagerDataService.getConstants(ConstantsPO po) | | | | 从数据库中返回ConstantsPO对象 | |

OrganizationController和Organization的接口规范如下表所示。

**OrganizationController的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| OrganizationController.addOrganization | 语法 | public ResultMessage addOrganization (OrganizationVO vo) | | | |
| 前置条件 | 已创建一个Organization领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Organization领域对象的addOrganization方法 | | | |
| OrganizationController.delOrganization | 语法 | public ResultMessage delOrganization(OrganizationVO vo) | | | |
| 前置条件 | 已创建一个Organization领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Organization领域对象的delOrganization方法 | | | |
| OrganizationController.revOrganization | 语法 | public ResultMessage revOrganization (OrganizationVO vo) | | | |
| 前置条件 | 已创建一个Organization领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Organization领域对象的revOrganization方法 | | | |
| OrganizationController.getOrganization | 语法 | public OrganizationVO getOrganization (string name) | | | |
| 前置条件 | 已创建一个Organization领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Organization领域对象的getOrganization方法 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| Organization.addOrganization | | | | 加入一个机构 | |
| Organization.delOrganization | | | | 删除一个机构 | |
| Organization.revOrganization | | | | 修改一个机构 | |
| Organization.getOrganization | | | | 返回一个机构 | |

**Organization的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| Organization.addOrganization | 语法 | public ResultMessage addOrganization (OrganizationVO vo) | | | |
| 前置条件 | 完成新机构信息的输入 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Organization.delOrganization | 语法 | public ResultMessage delOrganization(OrganizationVO vo) | | | |
| 前置条件 | 确认修改一个机构信息 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Organization.revOrganization | 语法 | public ResultMessage revOrganization (OrganizationVO vo) | | | |
| 前置条件 | 完成机构信息的修改 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Organization.getOrganization | 语法 | public OrganizationVO getOrganization (string name) | | | |
| 前置条件 | 数据库中存在该机构 | | | |
| 后置条件 | 返回OrganizationVO对象 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| ManagerDataService.addOrganization(OrganizationPO po) | | | | 向数据库中插入OrganizationPO对象 | |
| ManagerDataService.deleteOrganization(OrganizationPO po) | | | | 从数据库中删除OrganizationPO对象 | |
| ManagerDataService.updateOrganization(OrganizationPO po) | | | | 从数据库中修改更新某个OrganizationPO对象 | |
| ManagerDataService.getOrganization(String name) | | | | 从数据库中返回OrganizationPO对象 | |

StaffController和Staff的接口规范如下表所示。

**StaffController的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| StaffController.addStaff | 语法 | public ResultMessage addStaff (StaffVO vo) | | | |
| 前置条件 | 已创建一个Staff领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Staff领域对象的addStaff方法 | | | |
| StaffController.delStaff | 语法 | public ResultMessage delStaff(StaffVO vo) | | | |
| 前置条件 | 已创建一个Staff领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Staff领域对象的delStaff方法 | | | |
| StaffController.revStaff | 语法 | public ResultMessage revStaff (StaffVO vo) | | | |
| 前置条件 | 已创建一个Staff领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Staff领域对象的revStaff方法 | | | |
| StaffController.getStaff | 语法 | public StaffVO getStaff (string name) | | | |
| 前置条件 | 已创建一个Staff领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Staff领域对象的getStaff方法 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| Staff.addStaff | | | | 加入一个人员 | |
| Staff.delStaff | | | | 删除一个人员 | |
| Staff.revStaff | | | | 修改一个人员 | |
| Staff.getStaff | | | | 返回一个人员 | |

**Staff的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| Staff.addStaff | 语法 | public ResultMessage addStaff (StaffVO vo) | | | |
| 前置条件 | 完成新人员信息的输入 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Staff.delStaff | 语法 | public ResultMessage delStaff(StaffVO vo) | | | |
| 前置条件 | 确认修改一个人员信息 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Staff.revStaff | 语法 | public ResultMessage revStaff (StaffVO vo) | | | |
| 前置条件 | 完成人员信息的修改 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Staff.getStaff | 语法 | public StaffVO getStaff (string name) | | | |
| 前置条件 | 数据库中存在该人员 | | | |
| 后置条件 | 返回StaffVO对象 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| ManagerDataService.addStaff(StaffPO po) | | | | 向数据库中插入StaffPO对象 | |
| ManagerDataService.deleteStaff(StaffPO po) | | | | 从数据库中删除StaffPO对象 | |
| ManagerDataService.updateStaff(StaffPO po) | | | | 从数据库中修改更新某个StaffPO对象 | |
| ManagerDataService.getStaff(String name) | | | | 从数据库中返回StaffPO对象 | |

SalaryController和Salary的接口规范如下表所示。

**SalaryController的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| SalaryController.addSalary | 语法 | public ResultMessage addSalary (SalaryVO vo) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的addSalary方法 | | | |
| SalaryController.delSalary | 语法 | public ResultMessage delSalary(SalaryVO vo) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的delSalary方法 | | | |
| SalaryController.revSalary | 语法 | public ResultMessage revSalary (SalaryVO vo) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的revSalary方法 | | | |
| SalaryController.getSalary | 语法 | public SalaryVO getSalary (Job job) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的getSalary方法 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| Salary.addSalary | | | | 加入一个薪水策略 | |
| Salary.delSalary | | | | 删除一个薪水策略 | |
| Salary.revSalary | | | | 修改一个薪水策略 | |
| Salary.getSalary | | | | 返回一个薪水策略 | |

**Salary的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| Salary.addSalary | 语法 | public ResultMessage addSalary (SalaryVO vo) | | | |
| 前置条件 | 完成新薪水策略信息的输入 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Salary.delSalary | 语法 | public ResultMessage delSalary(SalaryVO vo) | | | |
| 前置条件 | 确认修改一个薪水策略信息 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Salary.revSalary | 语法 | public ResultMessage revSalary (SalaryVO vo) | | | |
| 前置条件 | 完成薪水策略信息的修改 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Salary.getSalary | 语法 | public SalaryVO getSalary (Job job) | | | |
| 前置条件 | 数据库中存在该薪水策略 | | | |
| 后置条件 | 返回SalaryVO对象 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| ManagerDataService.addSalary(SalaryPO po) | | | | 向数据库中插入SalaryPO对象 | |
| ManagerDataService.deleteSalary(SalaryPO po) | | | | 从数据库中删除SalaryPO对象 | |
| ManagerDataService.updateSalary(SalaryPO po) | | | | 从数据库中修改更新某个SalaryPO对象 | |
| ManagerDataService.getSalary(SalaryPO po) | | | | 从数据库中返回SalaryPO对象 | |

SalaryController和Salary的接口规范如下表所示。

**SalaryController的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| SalaryController.addSalary | 语法 | public ResultMessage addSalary (SalaryVO vo) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的addSalary方法 | | | |
| SalaryController.delSalary | 语法 | public ResultMessage delSalary(SalaryVO vo) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的delSalary方法 | | | |
| SalaryController.revSalary | 语法 | public ResultMessage revSalary (SalaryVO vo) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的revSalary方法 | | | |
| SalaryController.getSalary | 语法 | public SalaryVO getSalary (Job job) | | | |
| 前置条件 | 已创建一个Salary领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用Salary领域对象的getSalary方法 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| Salary.addSalary | | | | 加入一个薪水策略 | |
| Salary.delSalary | | | | 删除一个薪水策略 | |
| Salary.revSalary | | | | 修改一个薪水策略 | |
| Salary.getSalary | | | | 返回一个薪水策略 | |

**Salary的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| Salary.addSalary | 语法 | public ResultMessage addSalary (SalaryVO vo) | | | |
| 前置条件 | 完成新薪水策略信息的输入 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Salary.delSalary | 语法 | public ResultMessage delSalary(SalaryVO vo) | | | |
| 前置条件 | 确认修改一个薪水策略信息 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Salary.revSalary | 语法 | public ResultMessage revSalary (SalaryVO vo) | | | |
| 前置条件 | 完成薪水策略信息的修改 | | | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | | | |
| Salary.getSalary | 语法 | public SalaryVO getSalary (Job job) | | | |
| 前置条件 | 数据库中存在该薪水策略 | | | |
| 后置条件 | 返回SalaryVO对象 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| ManagerDataService.addSalary(SalaryPO po) | | | | 向数据库中插入SalaryPO对象 | |
| ManagerDataService.deleteSalary(SalaryPO po) | | | | 从数据库中删除SalaryPO对象 | |
| ManagerDataService.updateSalary(SalaryPO po) | | | | 从数据库中修改更新某个SalaryPO对象 | |
| ManagerDataService.getSalary(SalaryPO po) | | | | 从数据库中返回SalaryPO对象 | |

SearchAccountController和SearchAccount的接口规范如下表所示。

**SearchAccountController的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| SearchAccountController. searchAccount | 语法 | public AccountVO searchAccount(String id) | | | |
| 前置条件 | 已创建一个SearchAccount领域对象，输入信息符合规范 | | | |
| 后置条件 | 调用SearchAccount领域对象的getSearchAccount方法 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| SearchAccount.getSearchAccount | | | | 返回一个薪水策略 | |

**SearchAccount的接口规范**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | | |
| SearchAccount. searchAccount | 语法 | public AccountVO searchAccount(String id) | | | |
| 前置条件 | 启动一次账户查询 | | | |
| 后置条件 | 返回AccountVO对象 | | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| accountdataservice.findAccount (String id) | | | | 从数据库中返回AccountPO对象 | |

StatisticsController和Statistics的接口规范如下表所示。

**StatisticsController的接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| StatisticsController. Statistics | 语法 | public StatisticsVO Statistics(long startTime,long endTime) | |
| 前置条件 | 已创建一个Statistics领域对象，输入符合规范 | |
| 后置条件 | 调用Statistics领域对象的Statistics方法 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 | |
| Statistics. Statistics(long startTime,long endTime) | | | 返回一个statisticsVO对象 | |

**Statistics的接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| Statistics. getStatistics | 语法 | public StatisticsVO Statistics(long startTime,long endTime) | |
| 前置条件 | 启动一次成本收益表查看 | |
| 后置条件 | 返回一个statisticsVO对象 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 | |
| balancedataservice.DebitnoteFormDataService. getDebitnoteForm | | | 返回所有的DebitnotePO | |
| balancedataservice.RecordpayFormDataService. getallRecordpayForm | | | 返回所有的RecordpayPO | |

ExamController和Exam的接口规范如下表所示。

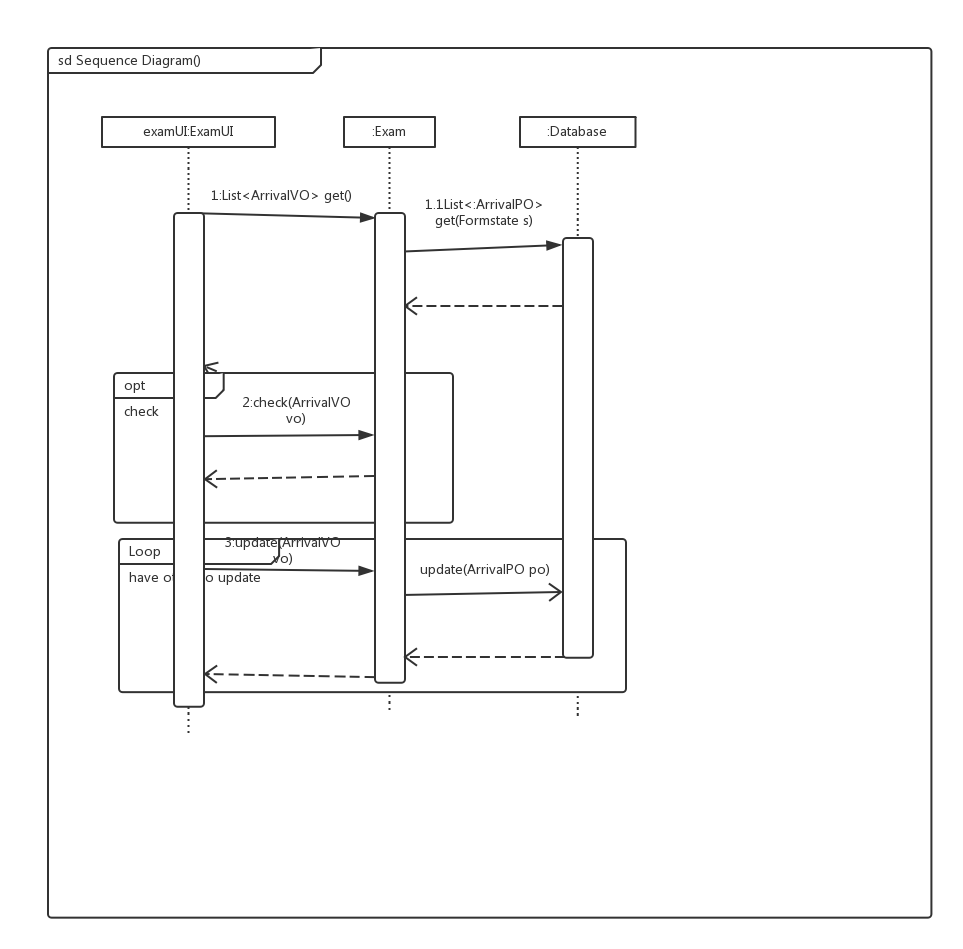
**ExamController的接口规范**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | |
| Examcontroller.getOrders | 语法 | public List<OrderExamVO> getOrders() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getOrders方法 | | |
| Examcontroller.OrdersExam | 语法 | public ResultMessage OrdersExam (List<OrderExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入符合规范 | | |
| 后置条件 | 调用Exam领域对象的OrdersExam方法 | | |
| Examcontroller.getHLForms | 语法 | public List<HLFormsExamVO> getHLForms () | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getHLForms方法 | | |
| Examcontroller. HLFormsExam | 语法 | public ResultMessage HLFormsExam (List< HLFormsExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入符合规范 | | |
| 后置条件 | 调用Exam领域对象的HLFormsExam方法 | | |
| Examcontroller.getCLForms | 语法 | public List<CLFormsExamVO> getCLForms () | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getCLForms方法 | | |
| Examcontroller. CLFormsExam | 语法 | public ResultMessage CLFormsExam (List< CLFormsExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的CLFormsExam方法 | | |
| Examcontroller.getGoodsRecevings | 语法 | public List<GoodsRecevingExamVO> getGoodsRecevings() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getGoodsRecevings方法 | | |
| Examcontroller. GoodsRecevingsExam | 语法 | public ResultMessage GoodsRecevingsExam (List<GoodsRecevingExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的GoodsRecevingsExam方法 | | |
| Examcontroller.getDeliverys | 语法 | public List<DeliveryExamVO> getDeliverys() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getDeliverys方法 | | |
| Examcontroller. DeliverysExam | 语法 | public ResultMessage DeliverysExam (List<DeliveryExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的DeliverysExam方法 | | |
| Examcontroller.getArrivals | 语法 | public List<ArrivalExamVO> getArrivals() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getArrivals方法 | | |
| Examcontroller.ArrivalsExam | 语法 | public ResultMessage ArrivalsExam (List<ArrivalExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的ArrivalsExam方法 | | |
| Examcontroller.getRecordtrans | 语法 | public List<RecordtranExamVO> getRecordtrans() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getRecordtrans方法 | | |
| Examcontroller.RecordtransExam | 语法 | public ResultMessage RecordtransExam (List<RecordtranExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的RecordtransExam方法 | | |
| Examcontroller.getInbounds | 语法 | public List<StockinExamVO> getInbounds() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getInbounds方法 | | |
| Examcontroller.InboundsExam | 语法 | public ResultMessage InboundsExam (List<StockinExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的InboundsExam方法 | | |
| Examcontroller.getStockouts | 语法 | public List<StockoutExamVO> getStockouts() | | |
| 前置条件 | 已创建一个Exam领域对象， | | |
| 后置条件 | 调用Exam领域对象的getStockouts方法 | | |
| Examcontroller.StockoutsExam | 语法 | public ResultMessage StockoutsExam (List<StockoutExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的StockoutsExam方法 | | |
| Examcontroller.getRecordcollects | 语法 | public List<RecordcollectExamVO> getRecordcollects() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getRecordcollects方法 | | |
| Examcontroller.RecordcollectsExam | 语法 | public ResultMessage RecordcollectsExam (List<RecordcollectExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的RecordcollectsExam方法 | | |
| Examcontroller.getDebitnotes | 语法 | public List<DebitnoteExamVO> getDebitnotes() | | |
| 前置条件 | 已创建一个Exam领域对象 | | |
| 后置条件 | 调用Exam领域对象的getDebitnotes方法 | | |
| Examcontroller.DebitnotesExam | 语法 | public ResultMessage DebitnotesExam (List<DebitnoteExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的DebitnotesExam方法 | | |
| Examcontroller.getRecordpays | 语法 | public List<RecordpayExamVO> getRecordpays() | | |
| 前置条件 | 数据库中存在收款单信息 | | |
| 后置条件 | 调用Exam领域对象的getRecordpays方法 | | |
| Examcontroller.RecordpaysExam | 语法 | public ResultMessage RecordpaysExam (List<RecordpayExamVO> volist) | | |
| 前置条件 | 已创建一个Exam领域对象，输入信息符合规范 | | |
| 后置条件 | 调用Exam领域对象的RecordpaysExam方法 | | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 | |
| Exam.getOrders() | | | | 返回List<OrderExamVO> | |
| Exam.OrdersExam(List<OrderExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getHLForms() | | | | 返回List<HLFormsExamVO> | |
| Exam. HLFormsExam(List< HLFormsExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getCLForms() | | | | 返回ListCHLFormsExamVO> | |
| Exam. CLFormsExam(List< CLFormsExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getGoodsRecevings() | | | | 返回List< GoodsRecevingExamVO > | |
| Exam.GoodsRecevingsExam(List<GoodsRecevingExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getDeliverys() | | | | 返回List< DeliveryExamVO VO> | |
| Exam.DeliverysExam (List<DeliveryExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getArrivals() | | | | 返回List< ArrivalExamVO > | |
| Exam.ArrivalsExam(List<ArrivalExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getRecordtrans() | | | | 返回List< ArrivalExamVO > | |
| Exam.RecordtransExam(List<ArrivalExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getInbounds() | | | | 返回List< StockinExamVO > | |
| Exam.InboundsExam(List<StockinExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getStockouts() | | | | 返回List< StockoutExamVO > | |
| Exam.StockoutsExam(List<StockoutExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getRecordcollects() | | | | 返回List< RecordcollectExamVO > | |
| Exam.RecordcollectsExam(List<RecordcollectExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getDebitnotes() | | | | 返回List< DebitnoteExamVO > | |
| Exam.DebitnotesExam(List<DebitnoteExamVO> volist) | | | | 存储审批后的单据 | |
| Exam.getRecordpays() | | | | 返回List< RecordpayExamVO > | |
| Exam.RecordpaysExam(List<RecordpayExamVO> volist) | | | | 存储审批后的单据 | |

**Exam的接口规范**

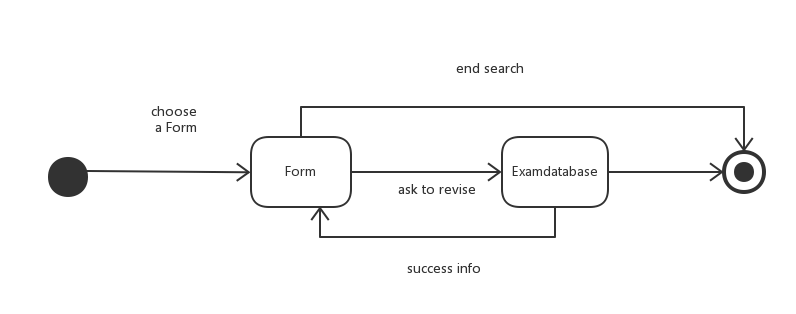
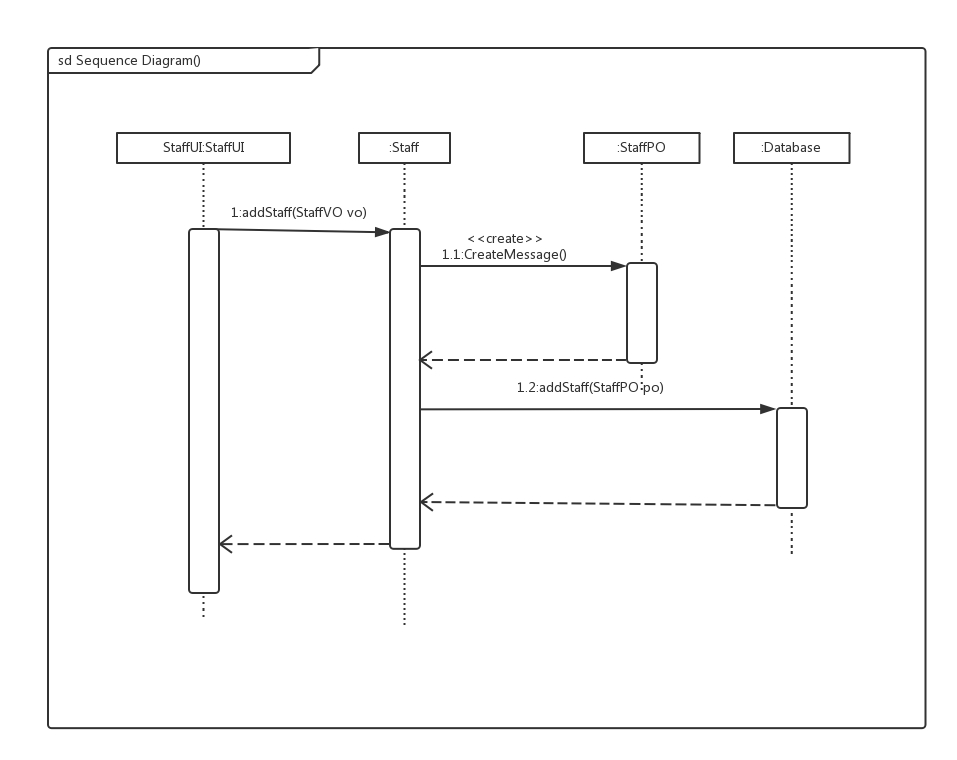
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | |
| Exam.getOrders | 语法 | public List<OrderExamVO> getOrders() | |
| 前置条件 | 开始订单查看 | |
| 后置条件 | 返回List<OrderExamVO> | |
| Exam.OrdersExam | 语法 | public ResultMessage OrdersExam (List<OrderExamVO> volist) | |
| 前置条件 | 开始订单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getHLForms | 语法 | public List<HLFormsExamVO> getHLForms () | |
| 前置条件 | 开始营业厅装车单查看 | |
| 后置条件 | 返回List< HLFormsExamVO > | |
| Exam. HLFormsExam | 语法 | public ResultMessage HLFormsExam (List< HLFormsExamVO> volist) | |
| 前置条件 | 开始营业厅装车单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getCLForms | 语法 | public List<CLFormsExamVO> getCLForms () | |
| 前置条件 | 开始中转中心装车单查看 | |
| 后置条件 | 返回List< CLFormsExamVO > | |
| Exam. CLFormsExam | 语法 | public ResultMessage CLFormsExam (List< CLFormsExamVO> volist) | |
| 前置条件 | 开始中转中心装车单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getGoodsRecevings | 语法 | public List<GoodsRecevingExamVO> getGoodsRecevings() | |
| 前置条件 | 开始收货单查看 | |
| 后置条件 | 返回List< GoodsRecevingExamVO > | |
| Exam. GoodsRecevingsExam | 语法 | public ResultMessage GoodsRecevingsExam (List<GoodsRecevingExamVO> volist) | |
| 前置条件 | 开始收货单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getDeliverys | 语法 | public List<DeliveryExamVO> getDeliverys() | |
| 前置条件 | 开始派件单查看 | |
| 后置条件 | 返回List< DeliveryExamVO > | |
| Exam. DeliverysExam | 语法 | public ResultMessage DeliverysExam (List<DeliveryExamVO> volist) | |
| 前置条件 | 开始派件单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getArrivals | 语法 | public List<ArrivalExamVO> getArrivals() | |
| 前置条件 | 开始到达单查看 | |
| 后置条件 | 返回List< ArrivalExamVO > | |
| Exam.ArrivalsExam | 语法 | public ResultMessage ArrivalsExam (List<ArrivalExamVO> volist) | |
| 前置条件 | 开始到达单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getRecordtrans | 语法 | public List<RecordtranExamVO> getRecordtrans() | |
| 前置条件 | 开始中转单查看 | |
| 后置条件 | 返回List< RecordtranExamVO > | |
| Exam.RecordtransExam | 语法 | public ResultMessage RecordtransExam (List<RecordtranExamVO> volist) | |
| 前置条件 | 开始中转单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getInbounds | 语法 | public List<StockinExamVO> getInbounds() | |
| 前置条件 | 开始入库单查看 | |
| 后置条件 | 返回List< StockinExamVO > | |
| Exam.InboundsExam | 语法 | public ResultMessage InboundsExam (List<StockinExamVO> volist) | |
| 前置条件 | 开始入库单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getStockouts | 语法 | public List<StockoutExamVO> getStockouts() | |
| 前置条件 | 开始出库单查看 | |
| 后置条件 | 返回List< StockoutExamVO > | |
| Exam.StockoutsExam | 语法 | public ResultMessage StockoutsExam (List<StockoutExamVO> volist) | |
| 前置条件 | 开始出库单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getRecordcollects | 语法 | public List<RecordcollectExamVO> getRecordcollects() | |
| 前置条件 | 开始收款单查看 | |
| 后置条件 | 返回List< RecordcollectExamVO > | |
| Exam.RecordcollectsExam | 语法 | public ResultMessage RecordcollectsExam (List<RecordcollectExamVO> volist) | |
| 前置条件 | 开始收款单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getDebitnotes | 语法 | public List<DebitnoteExamVO> getDebitnotes() | |
| 前置条件 | 开始收款单查看 | |
| 后置条件 | 返回List< DebitnoteExamVO > | |
| Exam.DebitnotesExam | 语法 | public ResultMessage DebitnotesExam (List<DebitnoteExamVO> volist) | |
| 前置条件 | 开始收款单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| Exam.getRecordpays | 语法 | public List<RecordpayExamVO> getRecordpays() | |
| 前置条件 | 开始付款单查看 | |
| 后置条件 | 返回List< RecordpayExamVO > | |
| Exam.RecordpaysExam | 语法 | public ResultMessage RecordpaysExam (List<RecordpayExamVO> volist) | |
| 前置条件 | 开始付款单审批 | |
| 后置条件 | 持久化更新涉及的领域对象的数据 | |
| 需要的服务（需接口） | | | | |
| 服务名 | | | | 服务 |
| Managerdataservice.Examdataservice.getOrders() | | | | 返回List<OrderExamPO> |
| Managerdataservice.Examdataservice.getHLForms() | | | | 返回List<HLFormsExamPO> |
| Managerdataservice.Examdataservice.getCLForms() | | | | 返回ListCHLFormsExamPO> |
| Managerdataservice.Examdataservice.getGoodsRecevings() | | | | 返回List< GoodsRecevingExamPO > |
| Managerdataservice.Examdataservice.getDeliverys() | | | | 返回List< DeliveryExamPO PO> |
| Managerdataservice.Examdataservice.getArrivals() | | | | 返回List< ArrivalExamPO > |
| Managerdataservice.Examdataservice.getRecordtrans() | | | | 返回List< ArrivalExamPO > |
| Managerdataservice.Examdataservice.getInbounds() | | | | 返回List< StockinExamPO > |
| Managerdataservice.Examdataservice.getStockouts() | | | | 返回List< StockoutExamPO > |
| Managerdataservice.Examdataservice.getRecordcollects() | | | | 返回List< RecordcollectExamPO > |
| Managerdataservice.Examdataservice.getDebitnotes() | | | | 返回List< DebitnoteExamPO > |
| Managerdataservice.Examdataservice.getRecordpays() | | | | 返回List< RecordpayExamPO > |

（4）业务逻辑层的动态模型

下图为总经理审批单据时的顺序图。

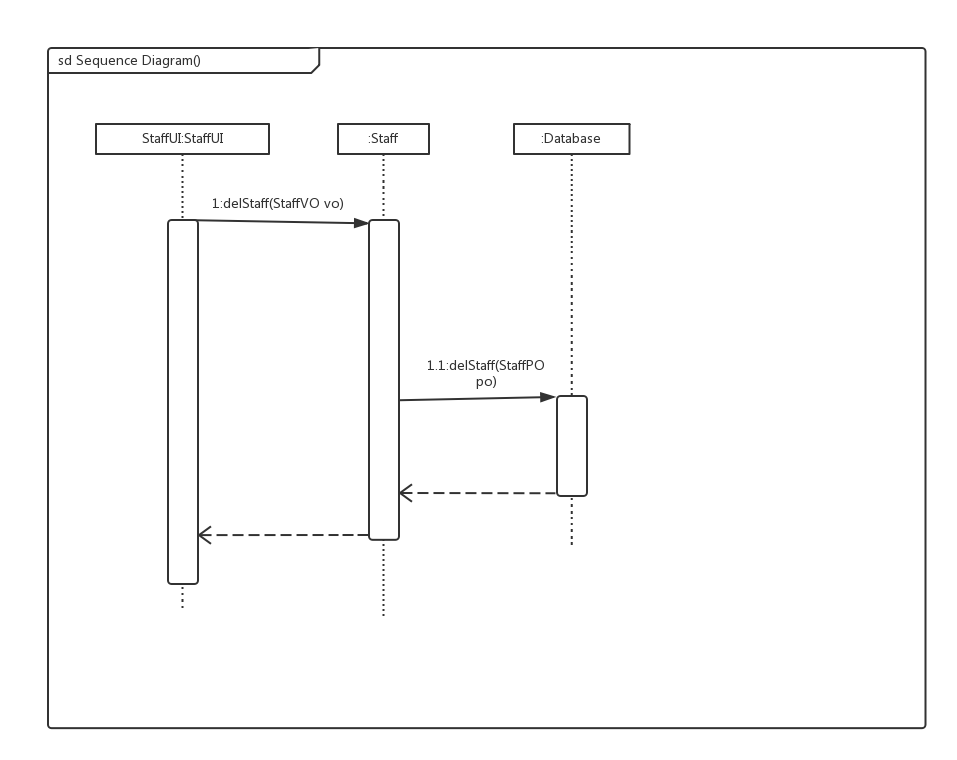
审批单据的顺序图

下图为总经理审批时的状态图



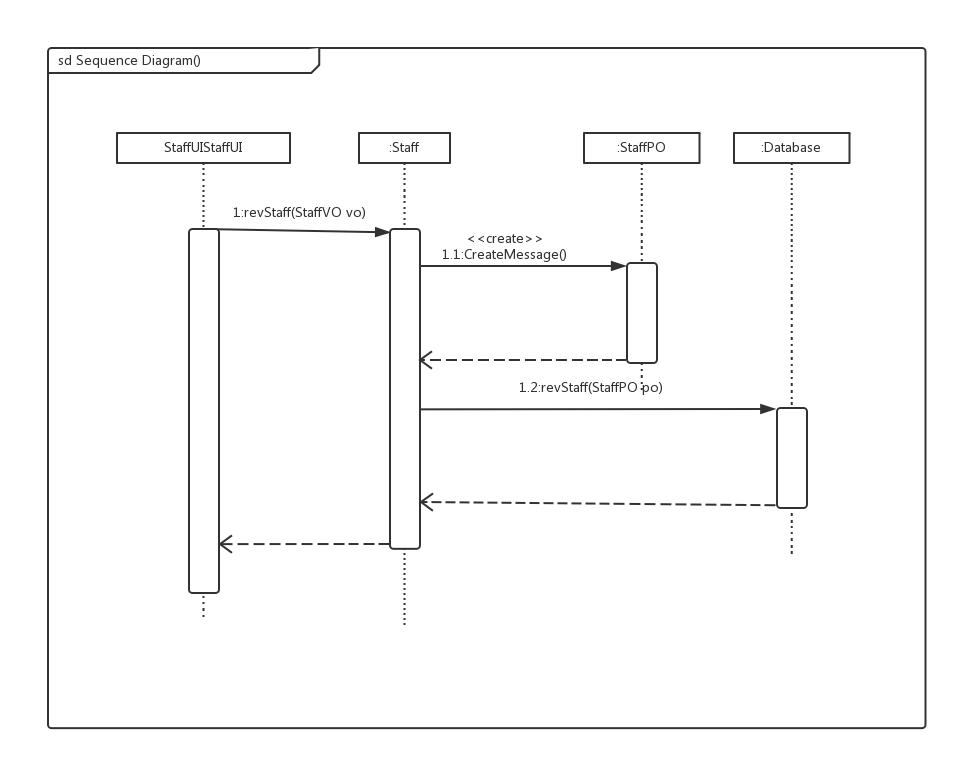
下图为总经理增加人员时的顺序图

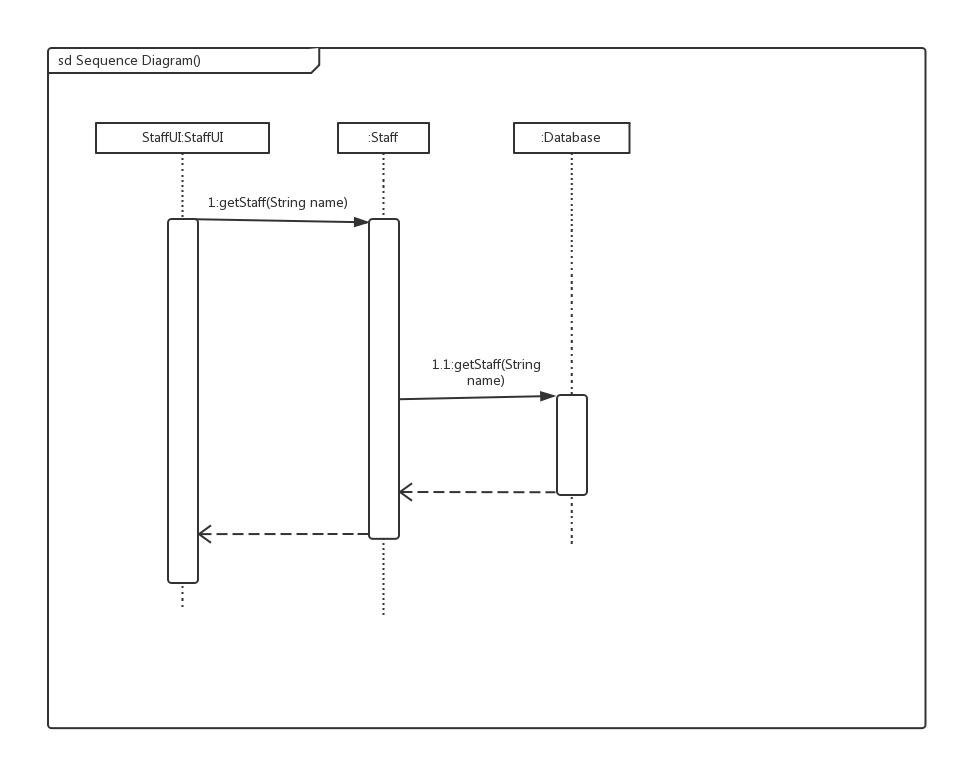
增加人员时的顺序图

下图为总经理删除人员时的顺序图

删除人员的顺序图

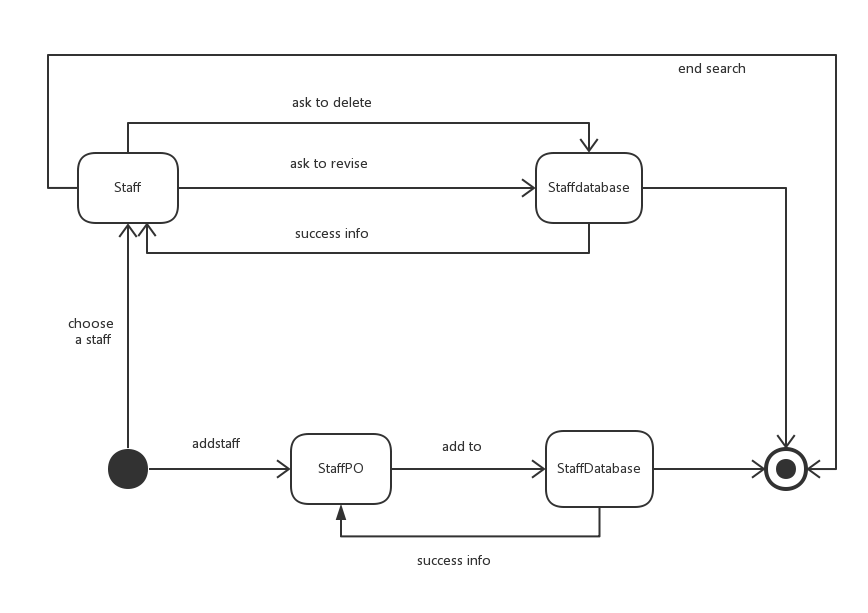
下图为总经理修改人员信息的顺序图

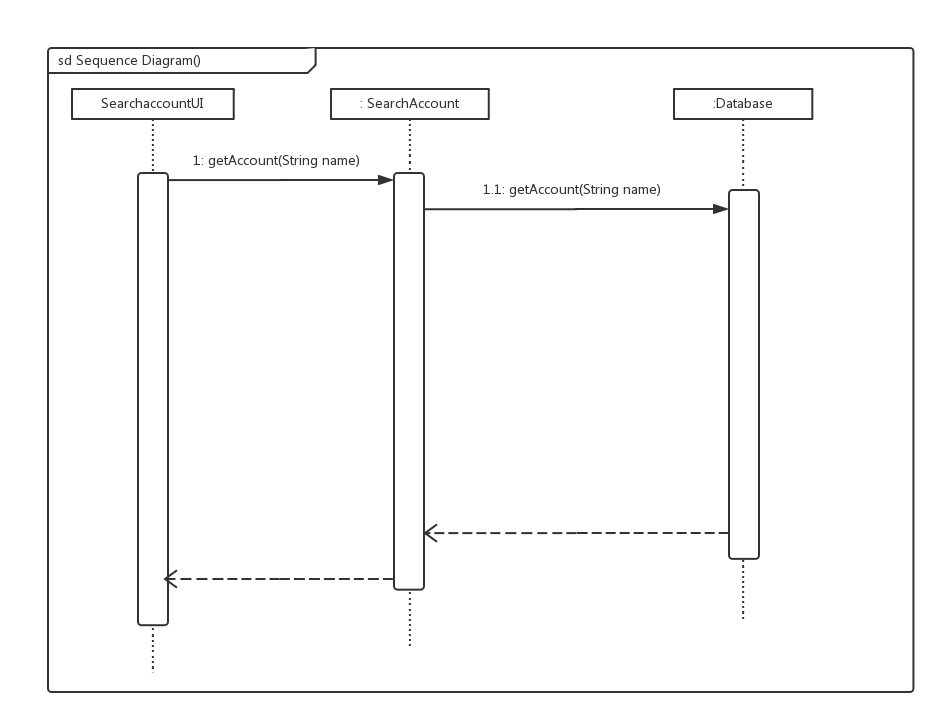
修改人员的顺序图

下图为总经理查询人员信息的顺序图

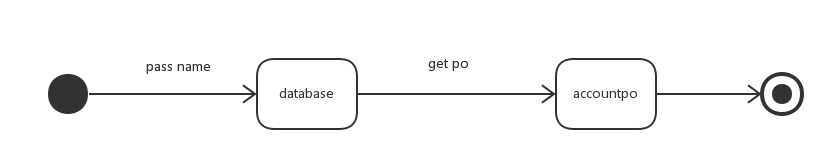
查询人员的顺序图

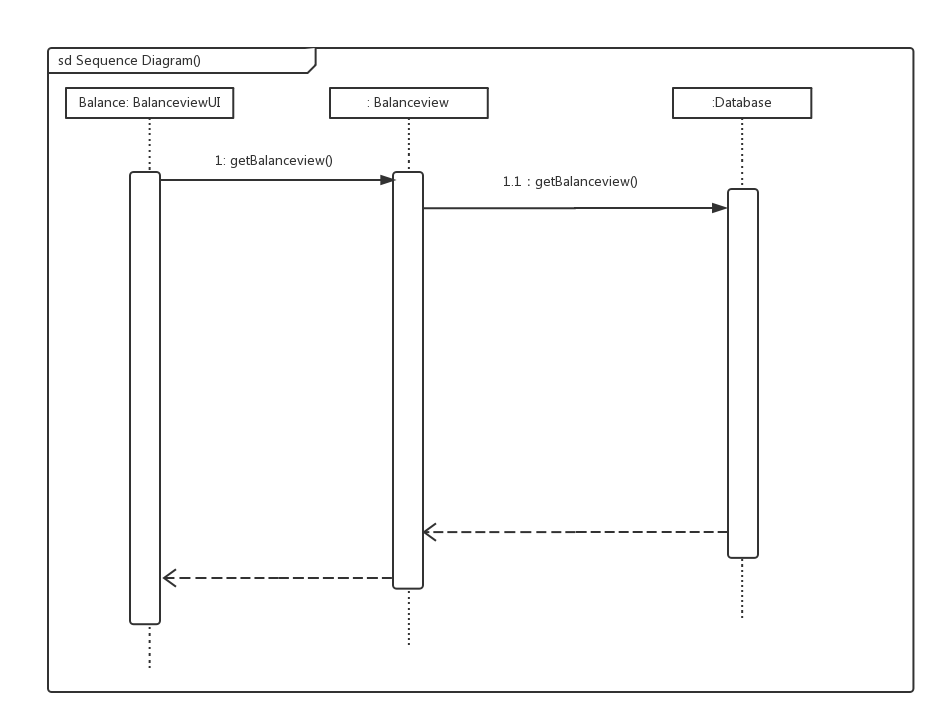
下图为总经理人员管理时的状态图

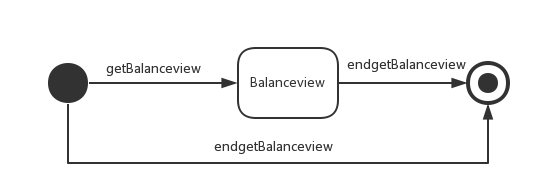
下图为总经理账户查询时的顺序图

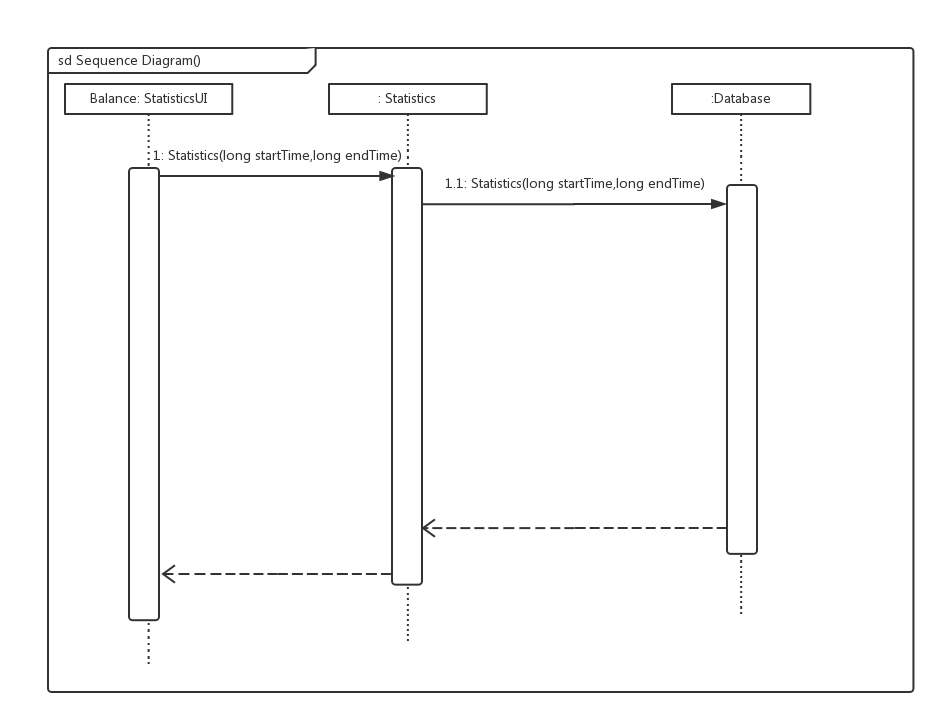


下图为总经理账户查询的状态图

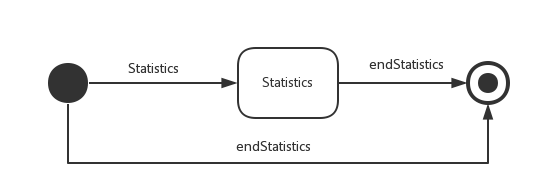
下图为总经理查看经营情况表时的顺序图

下图为总经理查看经营情况表时的顺序图

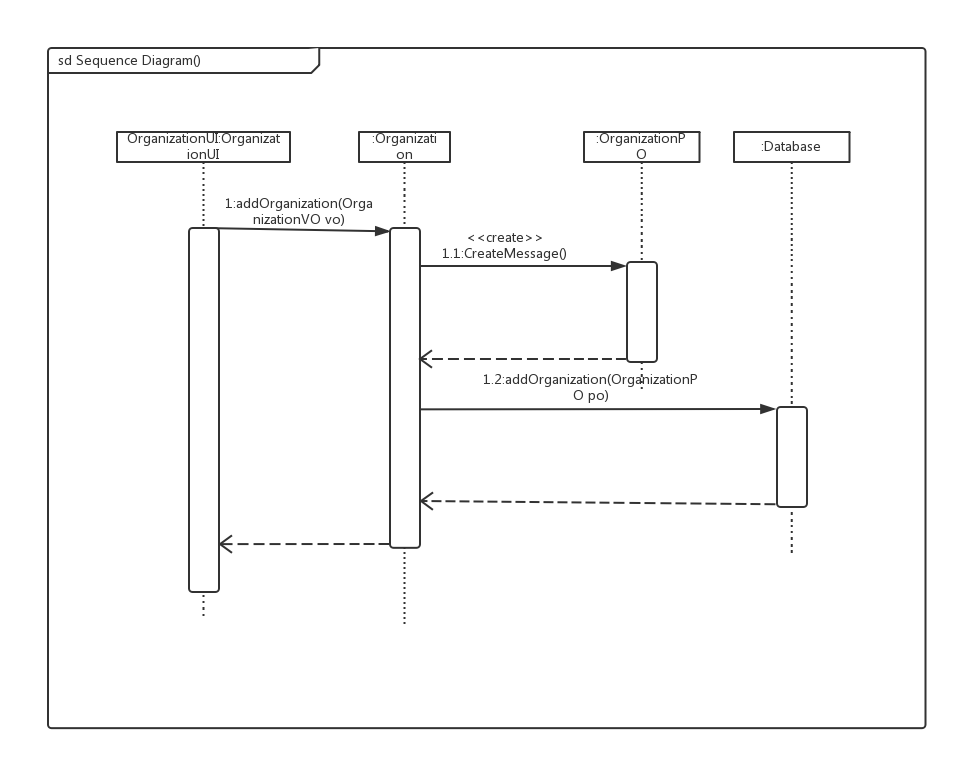
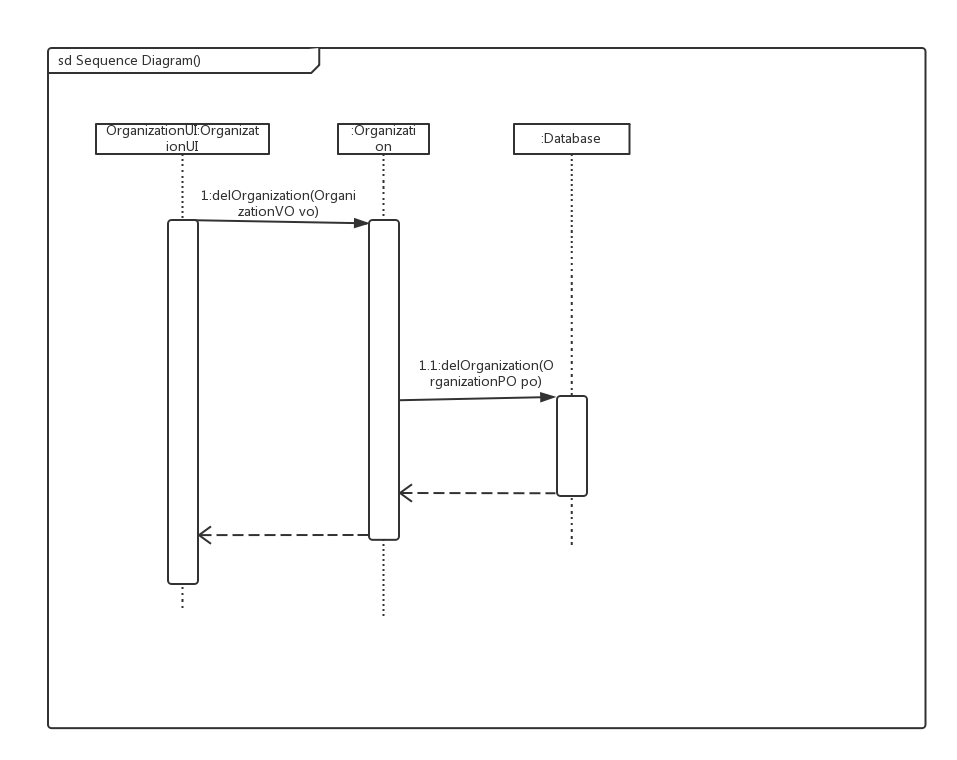
下图为总经理查看统计报表时的顺序图



下图为总经理查看统计报表时的状态图



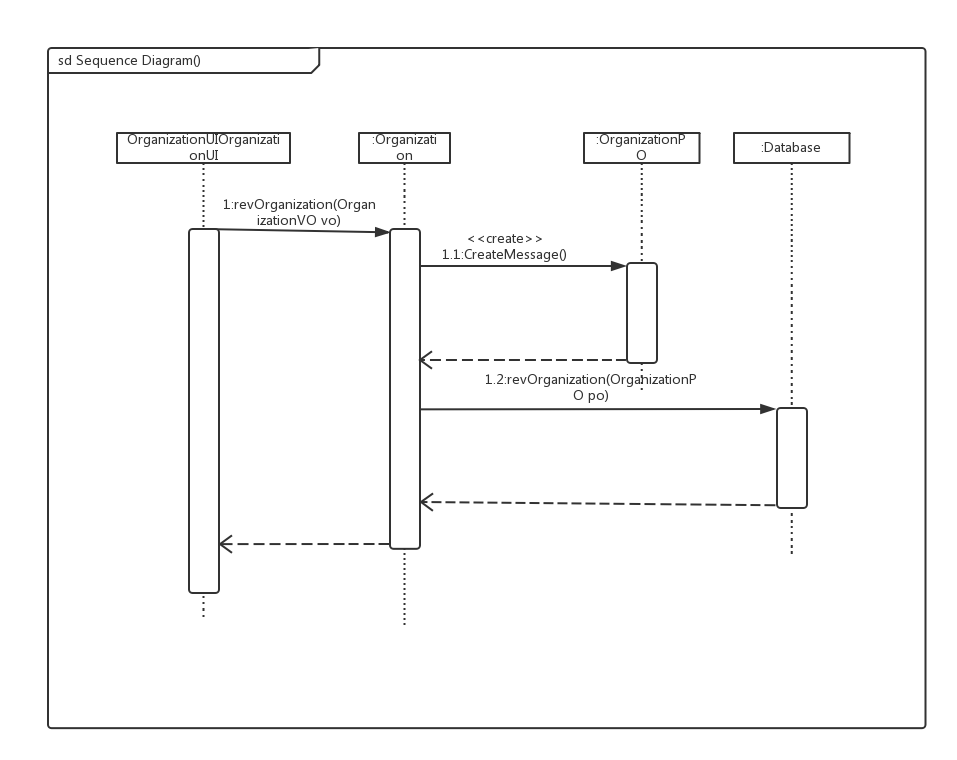
下图为总经理增加机构时的顺序图

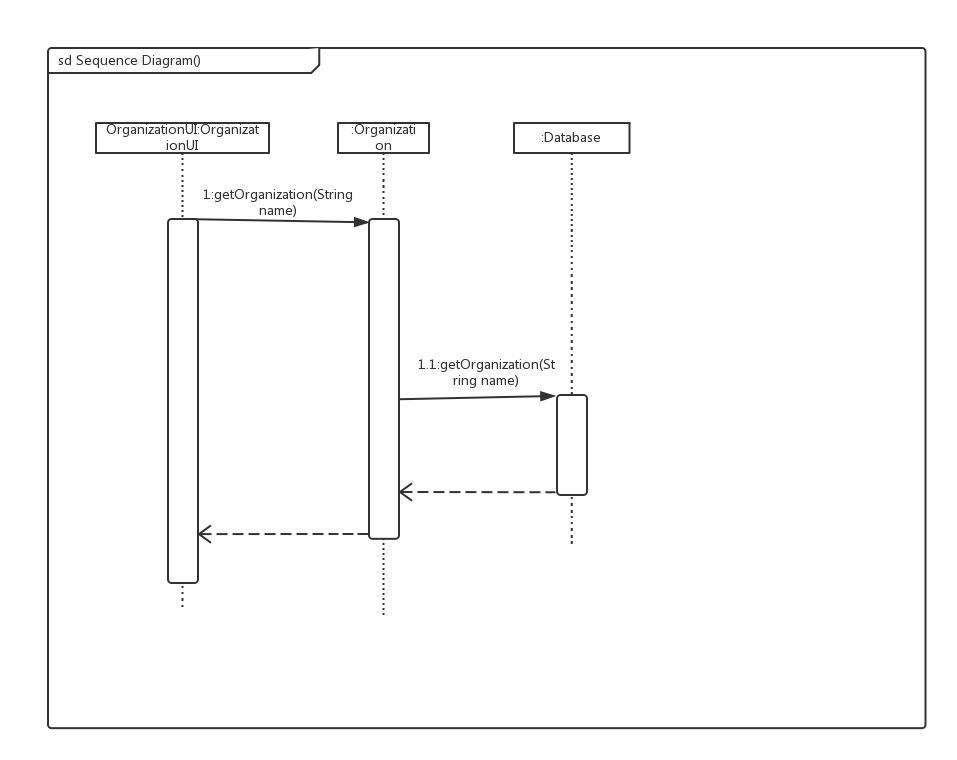
增加机构时的顺序图

下图为总经理删除机构时的顺序图

删除机构的顺序图

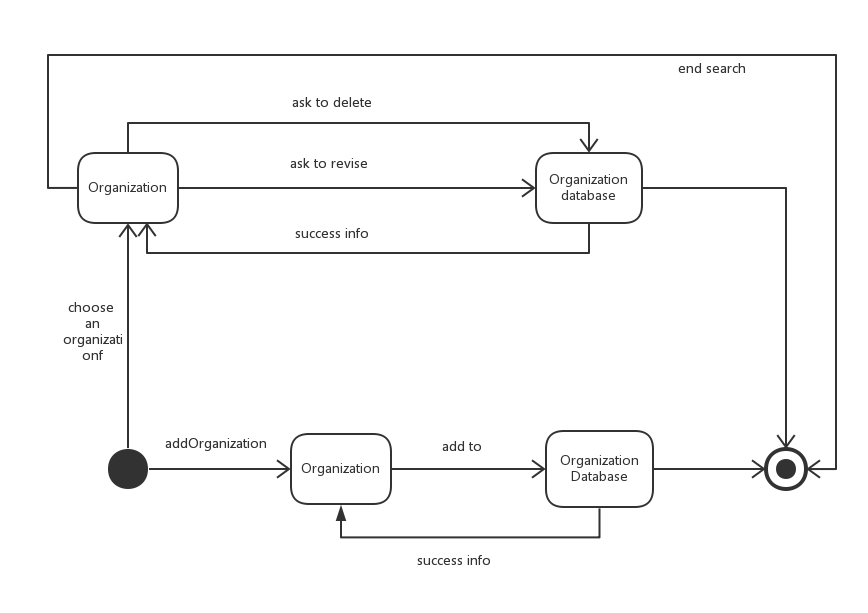
下图为总经理修改机构时的顺序图

修改机构时的顺序图

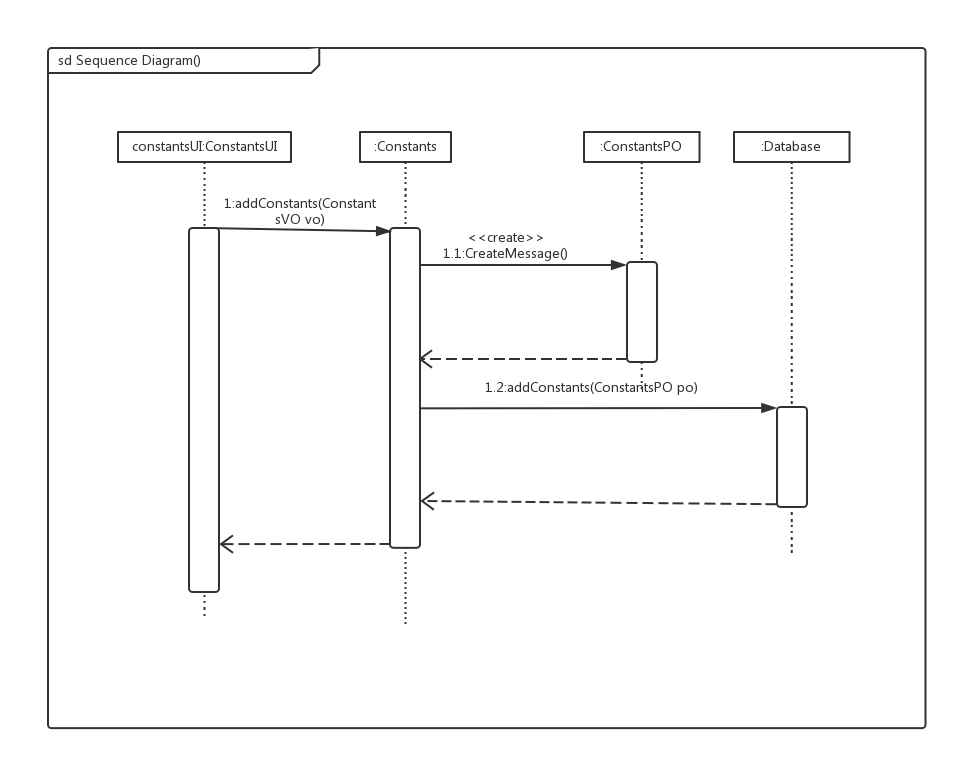
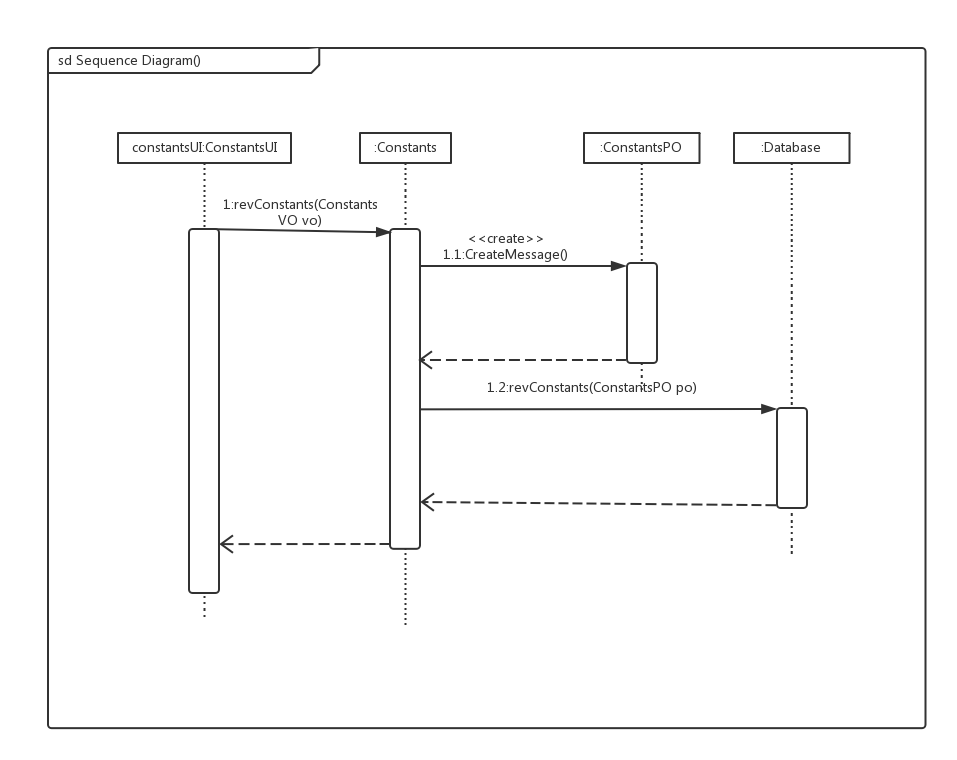
下图为总经理查询机构时的顺序图

查询机构时的顺序图

下图为总经理机构管理时的状态图



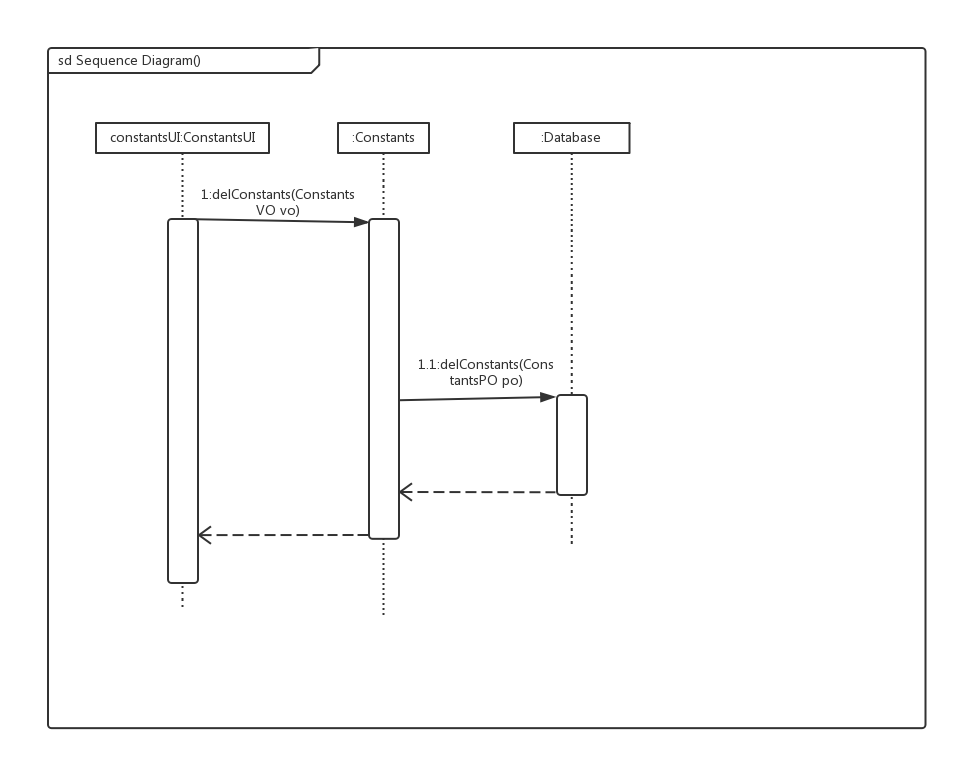
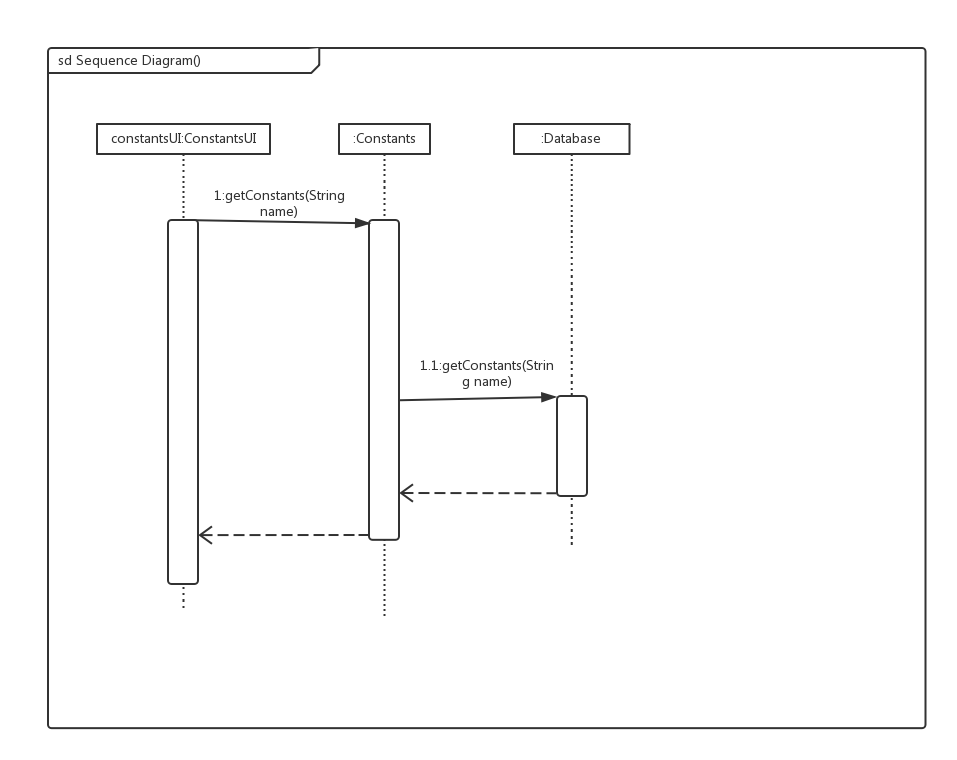
下图为总经理增加常量时的顺序图

下图增加常量的顺序图

下图为总经理修改常量时的顺序图

修改常量时的顺序图

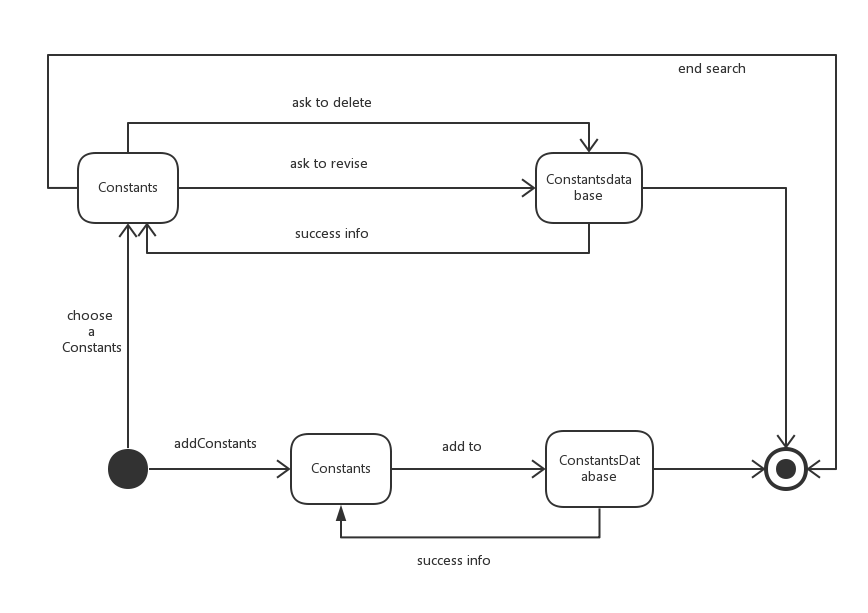
下图为删除常量时的顺序图

删除常量的顺序图

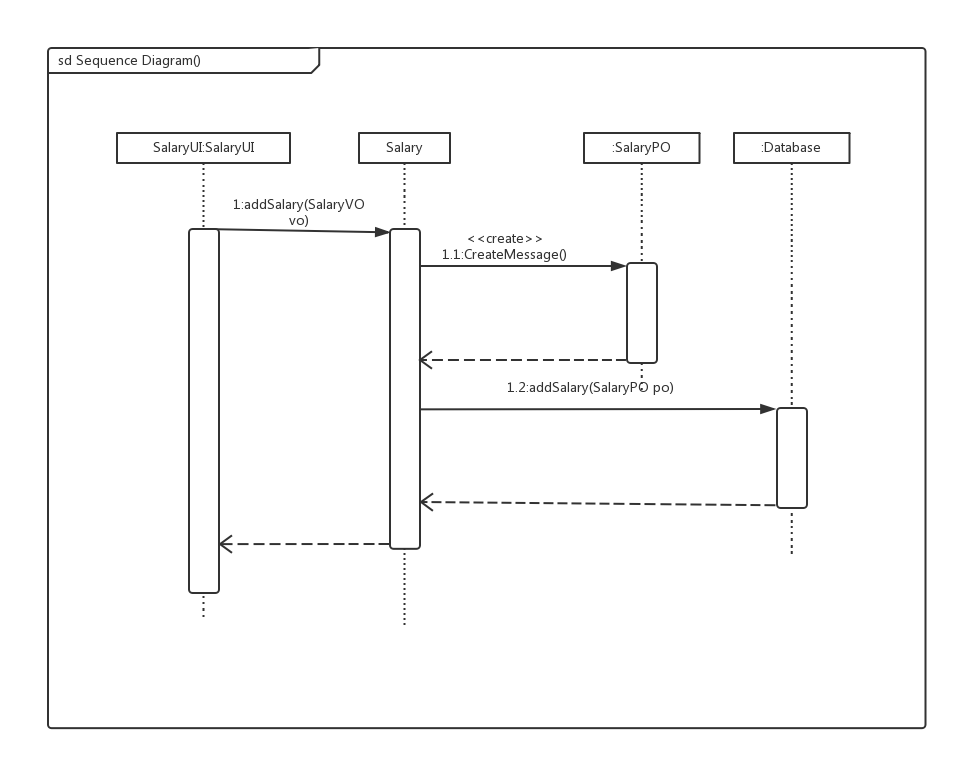
下图为总经理查询常量时的顺序图

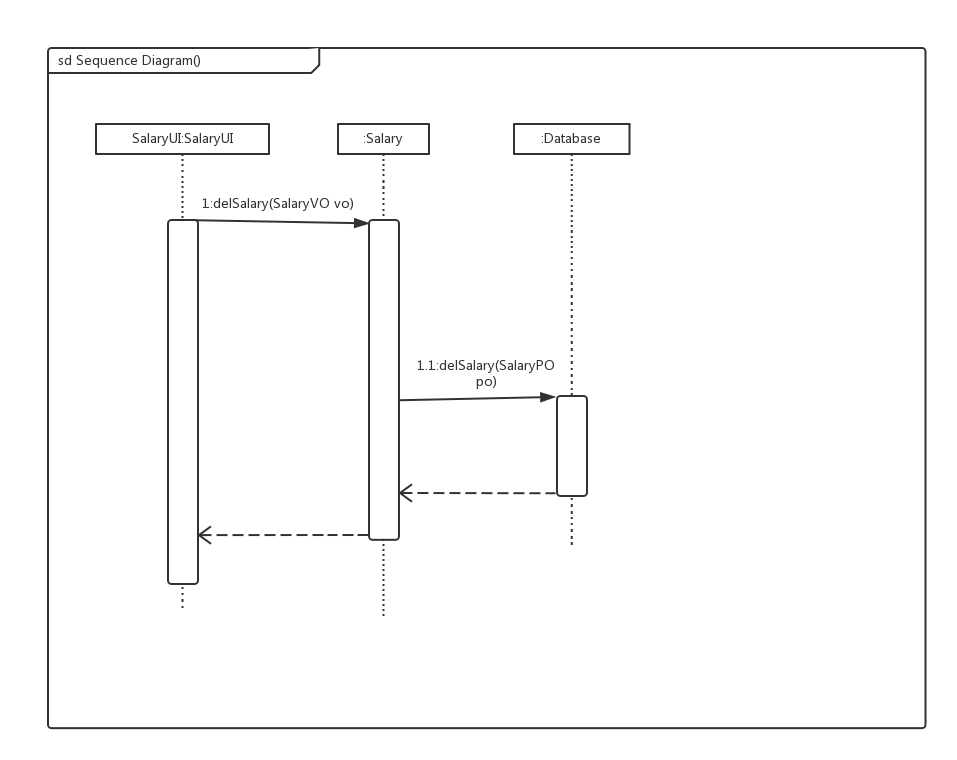
查询常量的顺序图

下图为总经理常量信息制定时的顺序图



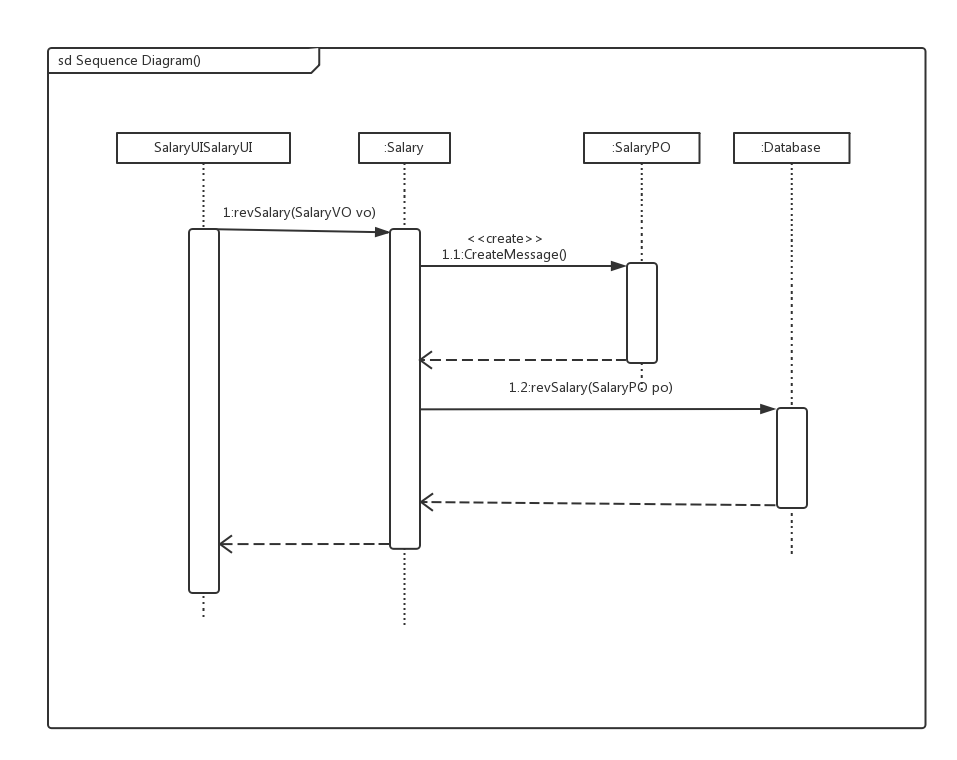
下图为总经理增加薪水策略的顺序图

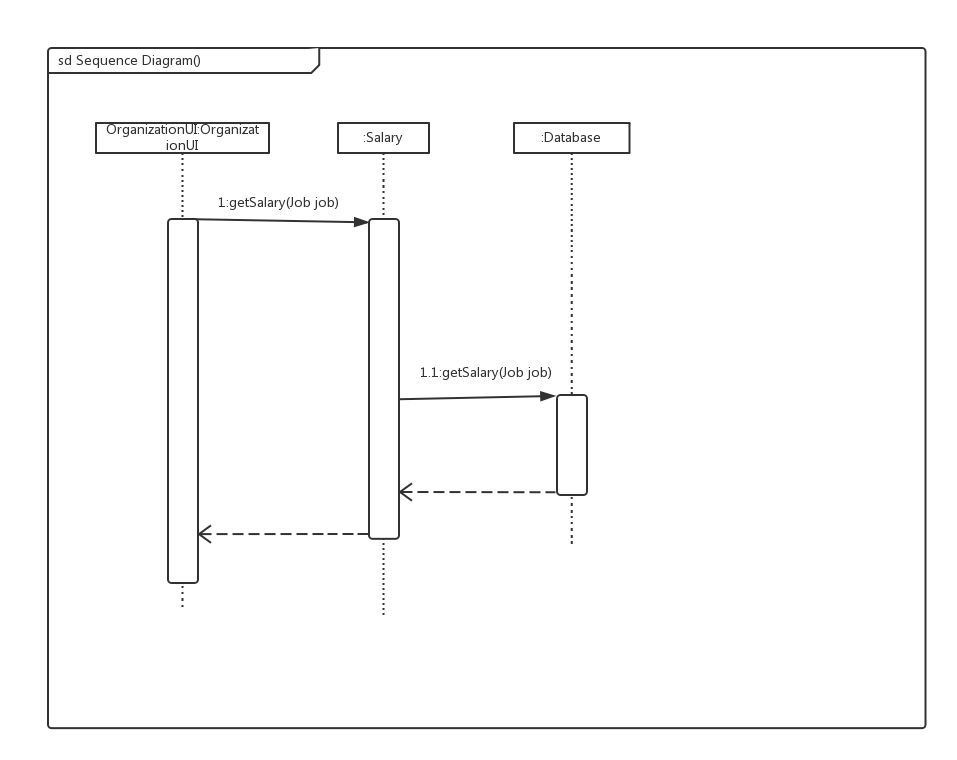
增加薪水策略的顺序图

下图为总经理删除薪水策略时的顺序图

删除薪水策略的顺序图

下图为总经理修改薪水策略的顺序图

修改薪水策略的顺序图

下图为总经理查询薪水策略的顺序图

查询薪水策略的顺序图

下图为总经理薪水策略制定时的状态图

