

基于OA办公系统的面向对象分析与设计

冀付军, 曹璐瑶

首都经济贸易大学管理工程学院软件工程专业, 北京

Email: 18331595798@163.com

收稿日期: 2021年3月28日; 录用日期: 2021年4月21日; 发布日期: 2021年4月28日

摘要

随着技术的不断提高, 现如今的企业也越来越注重系统分析和设计的思想, 其中面向对象分析与设计思想以其高适用性及易理解、更符合开发习惯等特点最为受到各大企业和团体的欢迎, 本文将全面介绍面向对象分析思想的相关概念及特性, 并结合OA办公系统, 以中小型企业为研究对象, 根据企业日常办公需求设计的办公自动化系统, 采用面向对象的方法对OA系统进行整体的分析和设计。本文主要从面向对象方法的方法、设计、UML分析方法等方面对面向对象进行系统的介绍, 并从OA系统的系统分析、系统设计、数据库设计等角度对面向对象思想的实践层面做进一步的说明。

关键词

面向对象分析, OA办公系统, UML分析

Object Oriented Analysis and Design of OA Office System

Fujun Ji, Luyao Cao

Major of Software Engineering, School of Management Engineering, Capital University of Economics and Business, Beijing

Email: 18331595798@163.com

Received: Mar. 28th, 2021; accepted: Apr. 21st, 2021; published: Apr. 28th, 2021

Abstract

With the continuous improvement of technology, today's enterprises are paying more and more attention to the idea of system analysis and design. Among them, the idea of object-oriented analysis and design is most popular among enterprises and groups because of its high applicability, easy to understand and more in line with the development habits. This paper will comprehensively

introduce the relevant concepts and characteristics of object-oriented analysis, and combine with OA office system to small and medium-sized enterprises as the research object, according to the enterprise daily office demand design of office automation system, using the object-oriented method to carry on the overall analysis and design of OA system. This paper mainly introduces the object-oriented method from the aspects of object-oriented method, design, UML analysis method and so on, and further explains the practical level of object-oriented thought from the system analysis, system design and database design of OA system.

Keywords

Object Oriented Analysis, OA Office System, UML Analysis

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

伴随着软件技术的快速发展,传统的开发方法逐渐的显露弊端,繁多的事务项很容易让人思维混乱,且很难重复使用,所以工作量极大,费用也相对较高,因此对应的程序设计语言也走上了进化的道路[1],从起初的机器语言、汇编语言直至各种结构化语言及后来的面向对象设计语言,在逐渐的演变过程中,这些语言越来越符合机器用语,也更符合开发习惯,可以复用,极大地降低了应用成本和时间,也使得代码的管理更加规范。

OA 办公系统在我国近几年迅速发展,尤其是随着疫情的到来,各大企业也开始纷纷完善自身的 OA 系统,但对于中小企业来说,因成本和模式问题,仍很难拥有适合自身业务的 OA 系统,同时设计难度较高,因此本文着重对面向对象这一思想方法进行分析,结合 OA 办公系统来形象化的运用面向对象应用技术,从而深刻理解面向对象的思想及应用。

2. 面向对象方法

2.1. 面向对象思想

对于一件事例或操作而言,通常有两种分析的思想方法,分别是面向对象和面向过程,这两者有着本质上的区别。

2.1.1. 面向过程

对于面向过程而言,分析问题的思路为将其拆解为不同的步骤,然后根据划分的步骤和过程来解决问题,是一种自上而下的设计方法,需要在解决问题前对其进行完整规划,这种逐步细化的方式会使得问题更易于理解和把控,但缺点是一旦投入开发,后期出现需求变动时难以更改和变动,对于解决实际问题来说不是十分灵活[2]。

2.1.2. 面向对象

面向对象则是采取抽象和封装的方式,将待解决问题抽象化出一般特点,分析出待研究问题中有哪些类和对象,然后综合分析这些对象应当具有哪些共有的属性和方法,分析类之间的关联和区别,在实际解决问题时,以实例化的方式实现各个类中的属性和方法。

主要采用的逻辑是从客观现实世界中出发, 基于存在的事例及事物角度, 尽可能运用人类的日常思维方式来全面的构造对应的系统, 从而反过来影响现实世界。

2.2. 面向对象分析与设计方法

面向对象的分析方法是在结合具体案例和示例的前提下, 对整个系统和业务, 采取面向对象的方法进行剖析, 强调运用人类在日常生活的逻辑思维中经常采用的思想方法与原则, 如抽象、分类, 继承、聚合、多态等。

2.2.1. 对象

对象是指待研究和解决的且具有高度抽象化的属性的事物及事件, 为各个基本单元的实体, 大量的对象实体构成整个系统, 因此针对对象进行实体具象化, 能极大的提高开发效率及开发质量[3]。

2.2.2. 类

类的概念是建立在对象基础之上, 它是大量的可抽象化对象的集合, 将类具体化之后就是对象, 采用数据结构的方式对其属性进行描述, 是某一类事物的属性与行为抽象的集合。总体来说是一个整体且抽象的概念。

2.2.3. 消息

消息是指对象与对象之间执行的操作信息, 也是对象和对象之间, 用作相互请求或者相互协作的唯一途径。

2.2.4. 面向对象的特性

第一是封装, 指将所分析问题抽象出共性, 并将这种共性及其属性、操作封装在一起, 让其成为一个完整的独立个体, 数据被装在内部, 通过抽象的数据类型保护起来, 不需要展现其中的细节, 在调用方法实例化时只需提供对外的接口即可实现与外部的联系, 完成解决问题方法的调用[4]。

第二个是继承, 即子类可以继承父类的属性和方法, 从而实现代码复用, 降低开发难度, 且能够极大程度缩短开发周期, 降低开发费用。

第三个是多态, 主要分为编译时多态和运行时多态, 编译时多态指的是方法的重载, 而运行时的多态指程序中定义的对象引用其指向的具体类型在运行期间才可以确定。

2.3. 面向对象 UML 分析

伴随面向对象建模方法而来的, 便是 UML 分析方法, 主要由事物、图和关系组成, 类似于软件领域的五线谱一般, 对于没有十分了解业务的开发人员来说也能一眼看懂, 帮助人们清晰快速的熟悉业务及开展工作, 且具有不易遗漏环节等优点, 所以至今被许多公司和团队用来分析业务, 从而更好地对项目进行把控。下面是一些主要的 UML 分析图类介绍[5]。

2.3.1. 用例图

主要用来描述作为一个外部观察者的视角对系统的印象, 强调人与系统的交互行为, 用例是为了完成某个工作或者达到某个目的一系列情节的总和, 角色是发动与这项工作或目的的人或事物。

2.3.2. 活动图

活动图通常是用来描述各用例及活动之间的约束关系, 是一种特殊的状态图, 用来表述过程基理、业务过程以及 workflows 的技术。描述对象活动的顺序关系所遵循的规则, 着重表现的是系统行为, 而非系统的处理过程[6]。

2.3.3. 类图

类图是静态的, 通过显示出系统所具有的类来表示系统, 可以显示系统所产生的影响, 类图存在三种关系:

- 1) 关联关系: 表示两种类实例间的关系, 强调两者之间存在必要关系, 通常在两个类之间用连线表示。
- 2) 聚合关系: 当某个类属于一个容器的特殊关系, 通常使用带菱形的连线, 指向具有整体性质的类。
- 3) 泛化关系: 指一个指向以其他类作为超类的继承连线, 通常用一个三角形指向超类。

2.3.4. 顺序图

顺序图通常描述对象间的交互作用, 将交互关系表示为一个二维图, 纵向为时间轴, 时间沿竖线向下延伸, 横向轴代表了在协作中各独立对象的类元角色, 类元角色通常用生命线表示, 当对象存在时, 角色用虚线表示, 当对象过程处于激活状态时, 时间线为双道线。

3. OA 办公系统分析

3.1. 研究背景

在国外, 办公自动化于 50 年代在美国和日本首先兴起, 这在起初只是具有电子数据处理(EDP)的簿记功能, 在 60 年代被管理信息系统(MIS)取代, 直到 70 年代后期才形成涉及多种技术的新型综合学科一办公自动化(OA)系统。到 80 年代, 国外的办公自动化已经得到了飞速发展, 许多著名的计算机软硬件公司都跻身于这一巨大的市场[7]。

对于我国的 OA 系统的发展, 已经从单纯的“传统办公”到“协同办公”软件, 再到“协同管理”软件, 乃至集多种软件功能于一身的“协同软件”。在协同管理理念的良好市场环境下, 我国的协同管理软件市场已经全面进入了快速发展期[8]。根据数据显示, 2016 年, OA 系统软件市场规模达到 89.4 亿元, 2017 年 OA 系统软件市场总规模超过了 103.7 亿元, 同比增长 15.99%, 连续三年增长率超过 15%, OA 办公软件市场已经进入稳定发展期。

在近些年, 随着大量的中小企业加入市场, OA 系统也经历了快速发展的阶段, 尤其是疫情期间, 有很多公司初期采取远程办公的方式, 这对于 OA 系统也有了更高的要求, 同时与此而来的, 市面上的厂商迎合这一时期, 快速产出适合企业多种办公方式的 OA 系统, 已适应绝大多数企业的基本日常需求, 解决了流程优化, 信息同步, 移动办公等问题, 但对于许多中小企业而言, 由于模式的不适用或价格的高昂原因, 仍在使用较为不完善的 OA 系统, 因此本文着重对于此类小企业的 OA 办公系统, 采取面向对象的分析方法对其进行分析和设计研究, 达到对此系统的全面了解及对面向对象分析方法的掌握[9]。

3.2. 系统特点

3.2.1. 管理员可以实时的观察员工的工作状态, 及时发现问题并解决问题

方便管理员工信息, 对员工新进入和员工离职都有合理、有序、统一的流程。辅助对员工绩效考评做出正确的分析。

3.2.2. 不同的角色拥有不同的权限

登入系统, 管理员和普通员工看到的菜单不同, 管理员可以根据员工职位的调整动态分配相应的权限, 员工可以随时查看个人信息, 比如是否存在异常打卡、是否有未完成任务、请假申请是否通过等。

3.2.3. 提供企业内部的沟通交流通道

登录系统, 打开在线聊天工具, 可以群聊, 也可以和指定的人聊。员工有什么问题可以及时反馈。

3.2.4. 人性化特点

根据中小企业实际情况, 实施合理的请假、出差、加班等的流程定义, 保证流程合理。

3.3. 初步调查

首先为合理分析系统, 对当前的中小企业系统进行调查和总结, 得出了目前中小型企业办公系统的现状:

- 1) 员工可以管理自己的基本信息, 只是页面响应缓慢, 页面不友好, 操作复杂, 用户体验不好。
- 2) 拥有简单的审批流程, 只是流程定义单一, 适用性低。
- 3) 大部分 OA 系统都实现了通用的功能, 只是缺少个性化, 市场占有率低。

3.4. 可行性分析与研究

3.4.1. 经济可行性

从成本上分析: 系统的成本费用主要包括软硬件的费用、服务器域名费用、系统设计开发费用和系统后期维护费用。系统操作简单, 不需要专门培训, 所以不存在培训费用。企业每个人都会配备一台电脑, 企业只需要将系统部署在服务器上, 员工通过域名即可访问系统。

从效益上分析: 使用 OA 办公系统简化了审批流程, 沟通交流通道畅通, 页面设计友好, 操作简单, 便于维护, 从而提高了工作效率, 节省了大量的人力物力财力。助力企业信息化, 提升企业的管理水平和综合竞争力, 无形中为企业带来更多经济效益[10]。

总体来说, 成本费用是固定的, 使用 OA 办公系统对企业带来的效益却是源源不断的。成本费用综合起来相对于整体效益而言只是很小的一部分, 所以经济上来说是可行的。

3.4.2. 技术可行性

对于 OA 系统来说, 可以采用 java 开发的方式, 系统所涉及的技术点包括: Activiti 工作流引擎、SSM 框架、AJAX 等。在技术上, 基本所有的功能都可实现, 技术要求均可达到。

Activiti 是一个开源的工作流引擎, 它实现了 BPMN 2.0 规范, 可以发布设计好的流程定义, 并通过 API 进行流程调度。其核心是基于 JAVA 的超快速、超稳定的 BPMN2.0 流程引擎, 强调流程服务的可嵌入性和可扩展性。将 Activiti 流程引擎应用于 OA 办公系统, 能够构建出功能丰富、轻便且高效的审批流程[11]。

SSM (Spring + SpringMVC + MyBatis) 框架由 Spring、SpringMVC、MyBatis 三个开源框架整合而成, 常用作简单的 web 项目的框架。其中 spring 是一个轻量级的控制反转(IoC)和面向切面(AOP)的容器框架。SpringMVC 分离了控制器、模型对象、分派器以及处理程序对象的角色, 这种分离让它们更容易进行定制。MyBatis 是一个支持普通 SQL 查询, 存储过程和高级映射的优秀持久层框架。

AJAX 最大的特点是页面无刷新, 使用异步方式与服务器通信, 不需要打断用户的操作, 具有更加迅速的响应能力, 在 OA 办公系统上使用 AJAX 可以很好的提高用户体验。[6]

可以结合以往 OA 系统的研发案例, 参考一些学习视频, 学习 Activiti5 工作流引擎的工作原理以及掌握 SSM 框架的搭建与配置, 熟悉 AJAX 的使用, 并结合目前 OA 办公系统的新需求, 所以 OA 办公系统的实现在技术上是可行的[12]。

3.4.3. 管理可行性

系统的使用对象主要是管理员和普通员工。系统界面友好, 操作简单, 对于一些稍微复杂的审批流程都进行了形象化处理, 并配以文字说明和提示, 辅助理解。系统在企业安装配置完成后, 员工只需要

通过网址进行访问即可。管理员对企业资源可以进行实时的监控和管理, 提高管理水平。因此 OA 办公系统的实现在管理方面是可行的[13]。

3.5. 业务分析

3.5.1. 用例图

系统角色主要分成三种, 系统管理员、普通管理员和员工, 其中系统管理员的用例图如图 1 所示, 系统管理员主要负责系统的维护、系统的基础配置和系统升级三个方面。

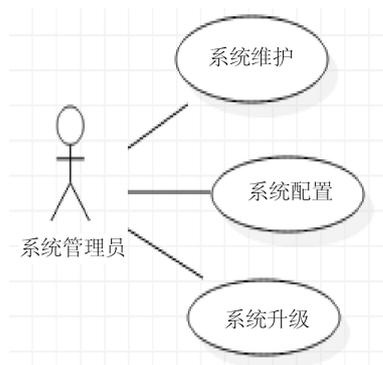


Figure 1. System administrator use case diagram

图 1. 系统管理员用例图

普通管理员和员工的用例图如图 2 所示。普通管理员与员工作为 OA 办公系统的使用者, 拥有共同的模块, 包括: 工作日程、工作总结、个人信息、消息管理、我的任务、备忘录、客户管理, 普通管理员与员工对这些模块都可以进行增删改查操作。

不同的模块包括, 普通管理员拥有组织结构、角色管理、员工管理、考勤管理、资料管理、会议、公告等模块, 而员工对于会议和公告只有查看功能, 没有编辑权限。

整个审批(请假、加班、出差等)流程的开始到结束, 是由管理员和员工共同完成的。对于考勤, 无论是管理员还是普通员工都要进行考勤签到, 并且只有管理员针对一些异常打卡情况进行备注处理, 月末进行考勤统计, 管理员可以导出考勤报表。对于公司组织结构、角色权限分配、员工职位调动、资料管理, 只有管理员拥有这些模块的权限。公告通知和会议记录, 管理员拥有增删改查所有权限, 普通员工只可以查看。

3.5.2. 活动图

在工作流中涉及的业务很多, 比如请假申请、加班申请、出差申请、报销申请等。

请假申请的活动图如图 3 所示。员工向部门负责人提交请假申请, 部门负责人进行审批, 部门负责人同意请假申请, 则继续向人事提交请假申请, 人事同意请假申请, 则请假申请通过, 流程结束。其中部门领导审批和人事审批只要有一个不同意, 则请假申请不予批准。

加班申请的活动图如图 4 所示。员工提出加班申请, 部门负责人审批, 部门负责人同意则人事审批, 人事审批同意, 则判断是正常加班还是异常加班, 如果是正常加班, 则无需总经理审批, 加班申请通过; 如果是异常加班, 则还要经过总经理审批, 总经理审批通过, 加班申请才被批准。

出差申请的活动图如图 5 所示。员工提交出差申请, 部门负责人审批, 部门负责人同意, 则总经理审批, 不同意则流程结束, 总经理审批同意, 则人事部进行备案, 并且财务部预支差旅费, 流程结束, 总经理审批不同意, 则流程直接结束。

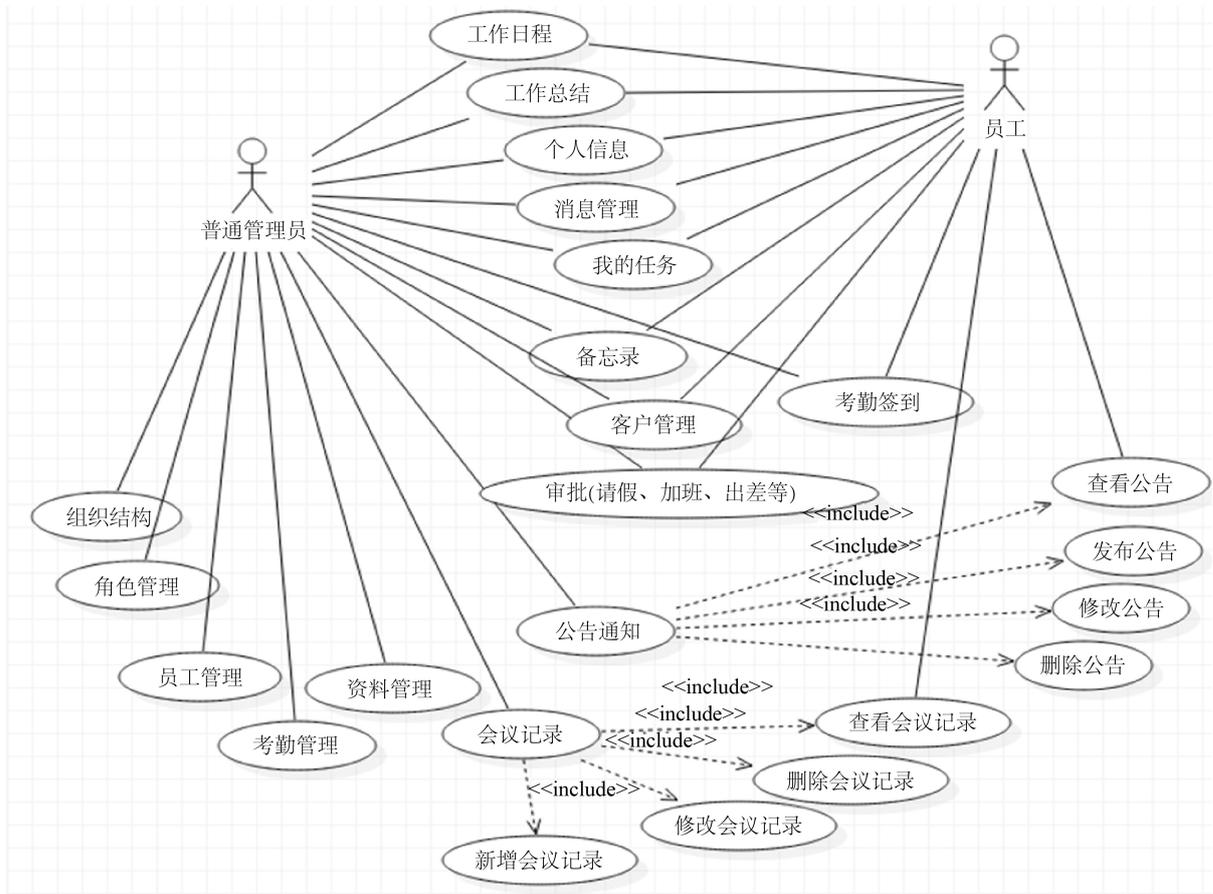


Figure 2. Use case diagram of ordinary administrators and employees
图 2. 普通管理员和员工的用例图

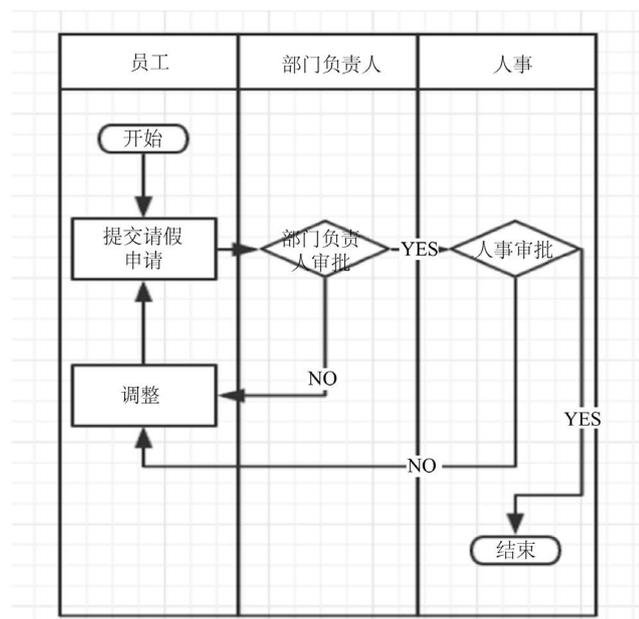


Figure 3. Diagram of leave application activity
图 3. 请假申请活动图

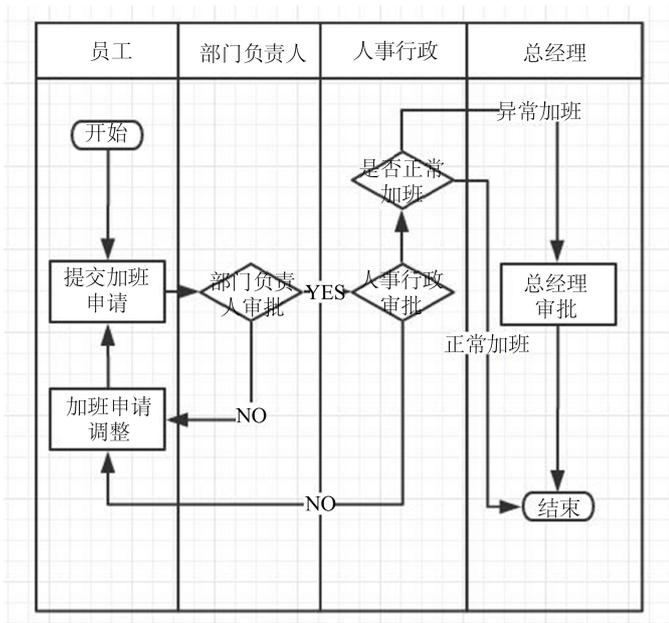


Figure 4. Activity diagram for overtime application
图 4. 加班申请的活动图

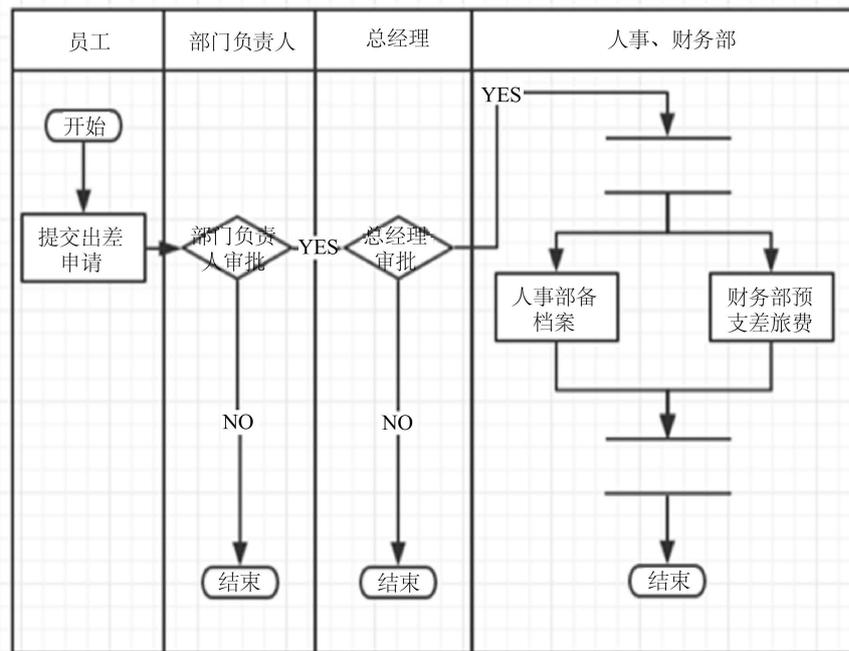


Figure 5. Activity diagram for business trip application
图 5. 出差申请的活动图

经费报销的活动图如图 6 所示。这个活动是其中最为复杂的一部分。

首先经办人填写报销单据，部门领导审批，部门领导同意，则财务预审判断是否超出预算，超出预算则需要部门主管、总会计师和总经理三级审批，没有超出预算，则直接到人事审批，在人事审批到财务审批的中间，会有金额限制的判断流程，判断是否超过 1000 元，超过 1000 元，则由财务经理审批，

没有超过 1000 元, 则由预算主管审批, 再判断是否超过 5000 元, 超过 5000 元需要总会计师审批, 然后是银行付款, 没有超过 5000 元, 直接银行付款, 最后由会计和出纳进行记账和归档操作。

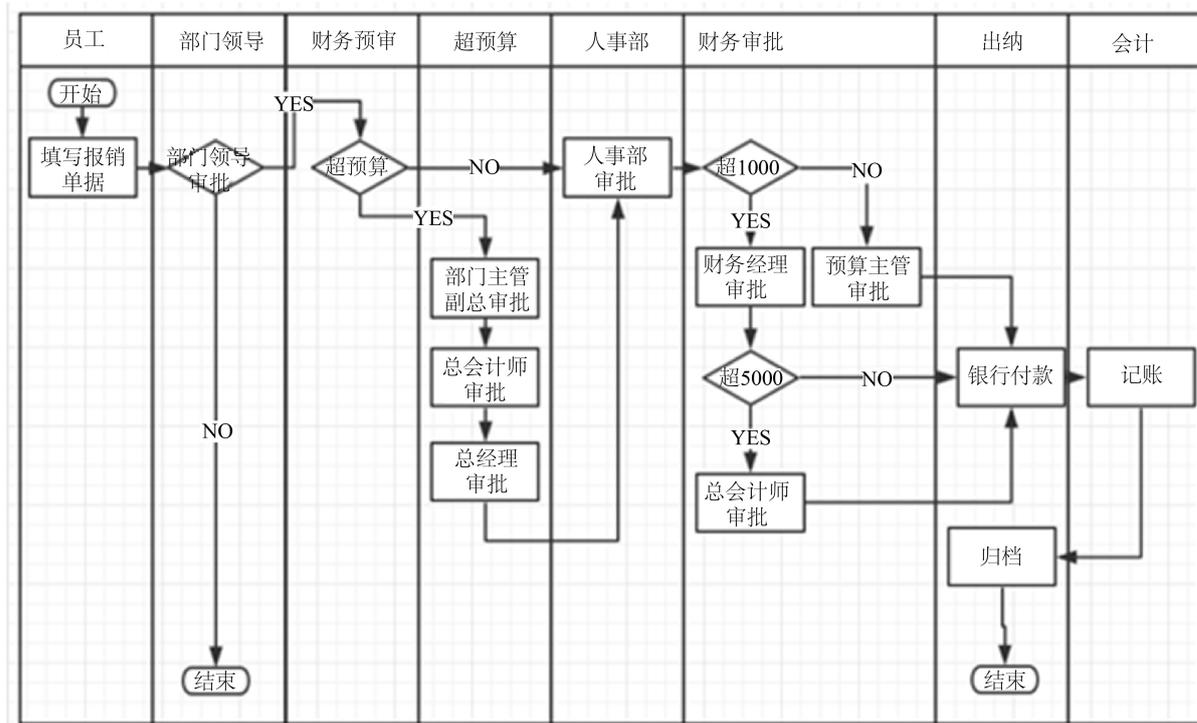


Figure 6. Activity diagram of expense reimbursement
图 6. 经费报销的活动图

通过对 OA 系统的用例图的分析, 成功定位该系统所拥有的角色、属性及角色之间的状态交互, 理清各角色的职能所在, 每个角色都为对象, 通过对每个对象的分析, 理解面向对象, 理解业务层面的角色表现。通过对活动图的分析, 很好的把握了实际主要业务场景的活动表现, 通过活动梳理流程, 并通过分析结果最终将重新反馈到系统, 达到实现的目的。

4. OA 办公系统设计

4.1. 系统设计目标

系统设计目标: 结合中小型企业办公流程, 设计一套功能完备, 流程合理, 界面友好, 与客户需求相吻合的 OA 办公系统, 突出协同办公、交流沟通方便和方便管理, 使用人性化, 突出企业应用的个性化, 并且系统能利用信息化带动企业管理进步, 提高企业竞争力。

4.2. 功能结构图

系统的功能结构图如图 7 所示, 通过之前的分析, 已经大致定位 OA 系统的基本功能方向, 因此结合实际业务场景后确定该 OA 系统的核心系统功能。

系统功能主要包括工作日程、工作总结、个人信息、审批、我的任务、考勤签到、备忘录、客户管理、组织结构、员工管理、资料管理、角色管理、公告通知、会议记录、考勤管理。其中审批又包括请假申请、加班申请、出差申请和报销。角色管理又包括权限分配和由于职位调动造成的角色更改。考勤

管理包括对异常打卡的处理和月末导出考勤报表。当然在实际应用中可能会涉及到更多的功能, 但在此处只对其中的关键模块做设计, 其他旁支功能在具体实践过程中可以再加以补充。

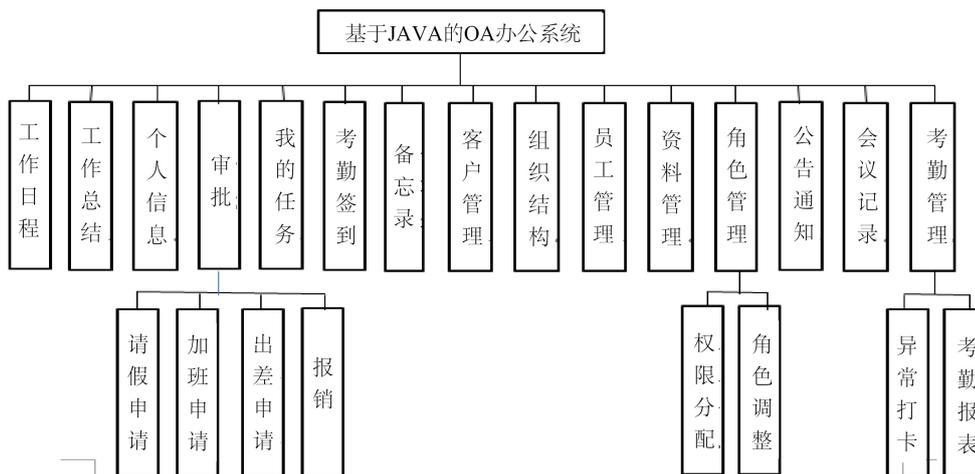


Figure 7. System function structure diagram
图 7. 系统功能结构图

4.3. 类图

类图可以更清晰直观的看到各个抽象化类别之间的关系, 同样也是构建数据库的基础, 在确定各个类之间关系的基础上, 才能更好的完成面向对象的开发工作。在此对 OA 系统的各个类进行分析的前提下, 构造了相关的类图, 如图 8 所示, 从而能对其有进一步的认识, 也为下一步数据库和开发做准备。

类图说明

这些类属于实体类, 所有类的属性都进行了私有化, 对外只提供一个 set、get 方法, 对内提供 toString 方法、有参构造方法、无参构造方法。下面是这些类的详细介绍。

1) Employee 是员工类, 它的属性包括员工编号(employee_ID)、用户名(userName)、密码(password)、真实姓名(name)、角色编号(role_ID)、最后登录时间(last_Login)、状态(status)、部门编号(ZD_ID)、IP 地址(IP)、邮箱(email)、联系电话(phone)。

2) Role 是角色类, 它的属性包括角色编号(role_ID)、角色名称(role_Name)、权限(rights)、添加权限(add_QX)、删除权限(del_QX)、修改权限(edit_QX)、查看权限(cha_QX)。

3) Dictionaries 是部门类, 它的属性包括部门编号(ZD_ID)、部门名称(name)、部门级别(level)、部门领导(manager)、上级部门编号(p_ID)。

4) Menu 是菜单类, 它的属性包括菜单编号(menu_ID)、菜单名称(menu_Name)、父类菜单编号(parent_ID)、菜单图标(menu_icon)、菜单类型(menu_type)、菜单地址(menu_url)。

5) Kaoqin 是考勤类, 它的属性包括考勤编号(kaoqin_ID)、员工编号(employee_ID)、签到时间(dktime)、上班或下班(work)、状态(state)、备注(remark)。

6) Memo 是备忘录类, 它的属性包括备忘录编号(memo_ID)、员工编号(employee_ID)、时间(datetime)、内容(content)。

7) Task 是任务类, 它的属性包括任务编号(task_ID)、员工编号(employee_ID)、标题(title)、内容(content)、发布时间(releasetime)、状态(state)。

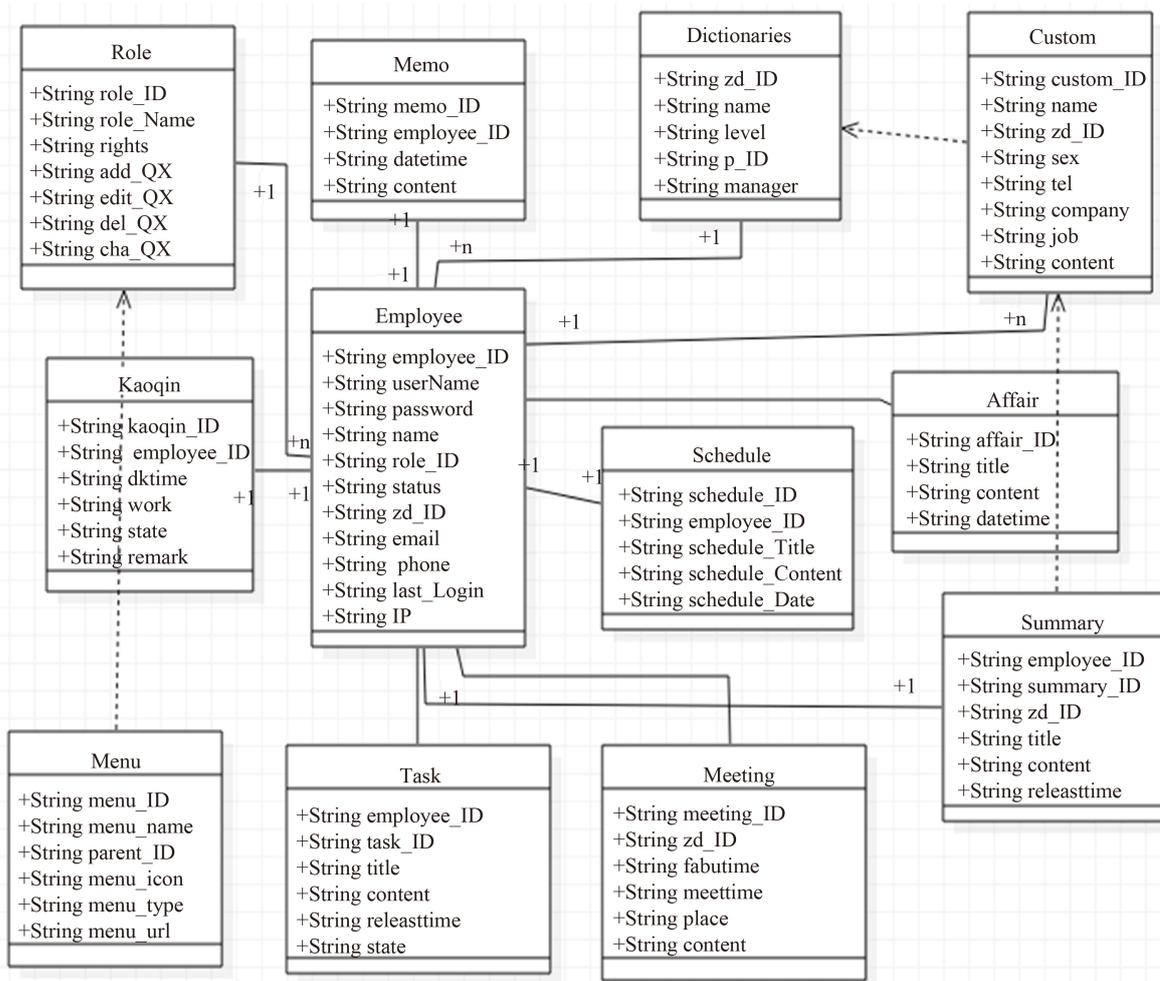


Figure 8. Entity class diagram

图 8. 实体类图

8) Schedule 是工作日程类, 它的属性包括工作日程编号(schedule_ID)、员工编号(employee_ID)、工作日程标题(schedule_Title)、工作日程内容(schedule_Content)、日期(schedule_Date)。

9) Meeting 是会议类, 它的属性包括会议编号(meeting_ID)、部门编号(ZD_ID)、发布时间(fabutime)、会议时间(meetime)、地点(place)、内容(content)。

10) Custom 是客户类, 它的属性包括客户编号(custom_ID)、姓名(name)、合作部门编号(ZD_ID)、性别(sex)、电话(tel)、所在公司(company)、职位(job)、合作内容(content)。

11) Affair 是公告通知类, 它的属性包括公告通知编号(memo_ID)、标题(title)、时间(datetime)、内容(content)。

12) Summary 是工作总结类, 它的属性包括工作总结编号(task_ID)、员工编号(employee_ID)、部门编号(ZD_ID)、标题(title)、内容(content)、发布时间(releastime)。

4.4. 顺序图

4.4.1. 管理员为员工分配权限以及发布任务图

如图 9 所示, 管理员登录成功以后, 进入主页面, 可以查看并修改员工信息, 为员工分配相应的权

限并且可以为员工发布任务, 除此之外还包括任务验收、任务查看等流程。员工有了权限以后, 登录进去, 只有查看权限, 可以查看自己的任务哪些没有做完, 完成任务并修改任务状态。

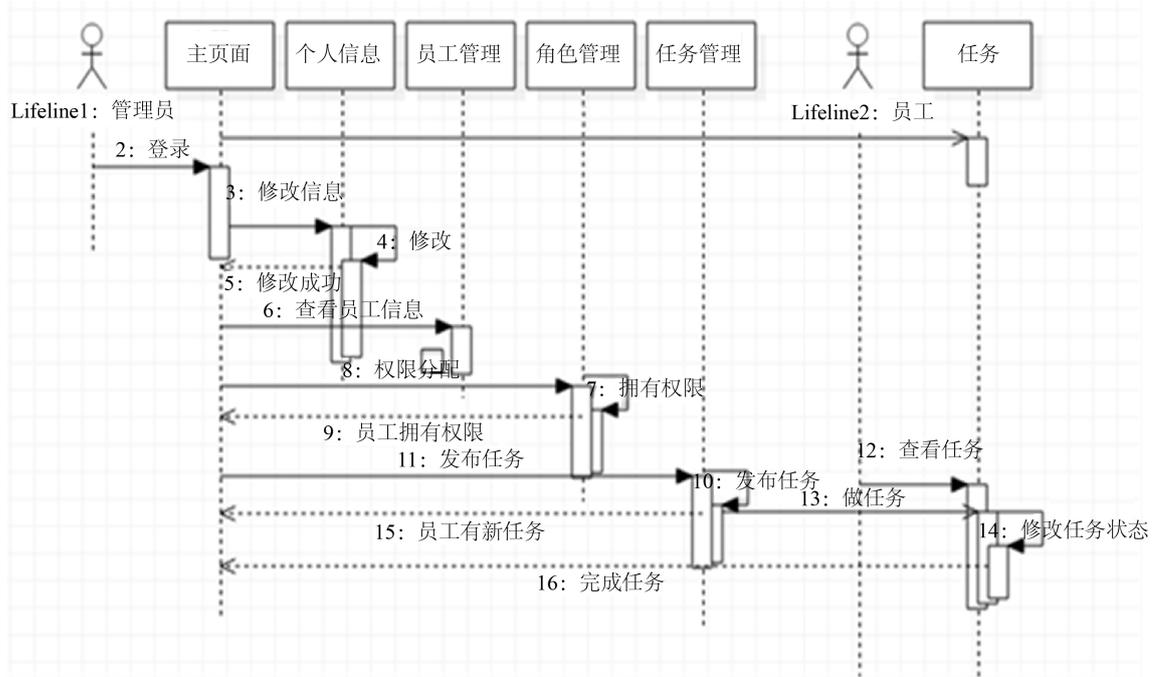


Figure 9. Diagram of the administrator assigning permissions to employees and publishing tasks
图 9. 管理员为员工分配权限以及发布任务图

4.4.2. 从人事部署流程到员工请假流程审批

如图 10 所示, 请假流程涉及角色也比较多, 因此需要严谨的后台逻辑做支撑。首先, 人事需要部署流程, 通过上传流程定义文件来完成部署流程, 上传请假流程规则, 然后员工添加请假申请并提交, 此时请假流程正式启动, 然后是部门领导审批, 部门领导审批通过后, 人事审批, 人事审批通过则批准请假。

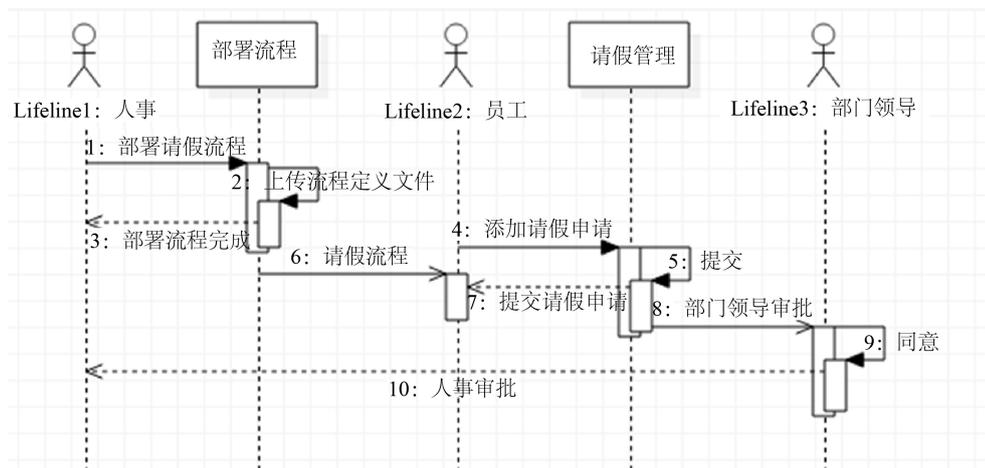


Figure 10. Personnel deployment process to employee leave process approval diagram
图 10. 人事部署流程到员工请假流程审批图

5. OA 系统数据库设计

在对业务进行梳理之后, 尝试进行了数据库的初步设计, 分别从概念数据模型和逻辑数据模型两个角度进行了设计, 物理模型并未深入展开。

5.1. 概念数据模型设计

数据库概念模型设计, 具体的可以用 E-R 图来表示。E-R 图的基本元素包括以下四种:

实体: 指客观存在着的实体, 对于员工表来说, 每一个员工就是一个实体;

属性: 是指实体的特征, 员工表中的员工编号、姓名等都可以看作实体的属性;

关系: 指的是某一个实体或某多个实体之间的关联关系, 如部门与员工之间是一对多的关系;

关系实体: 表示有关联关系的具体实体。

相关 E-R 图如图 11 所示: 员工与部门是多对一的关系, 员工与角色是多对一的关系。

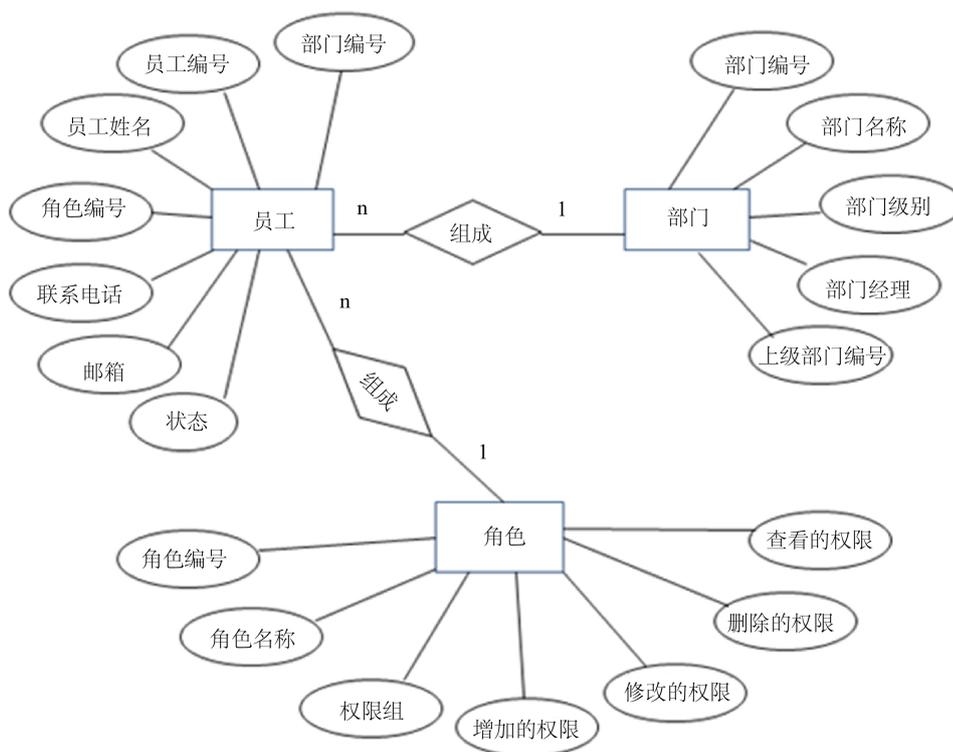


Figure 11. E-R diagram of main entities

图 11. 主要实体的 E-R 图

如下图 12 是菜单的实体图, 包括父菜单编号、菜单编号、菜单名称、菜单地址、菜单类型、菜单图标等属性。

如下图 13 是考勤实体图, 包括员工编号、签到时间、上班或下班、状态、备注、考勤编号等属性。

如下图 14 是备忘录实体图, 包括备忘录编号、员工编号、时间、内容等属性。

如下图 15 是工作日程实体图, 包括工作日程编号、日程标题、日程内容、日期等属性。

如下图 16 是公告通知实体图, 包括公告通知编号、标题、内容、发布时间等属性。

如下图 17 是会议记录实体图, 包括会议记录编号、会议内容、会议地点、会议时间、发布时间、部门编号等属性。

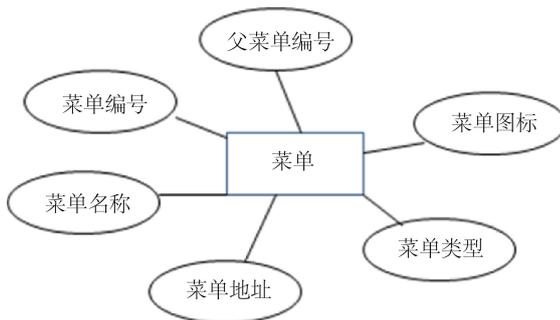


Figure 12. Menu entity diagram
图 12. 菜单实体图

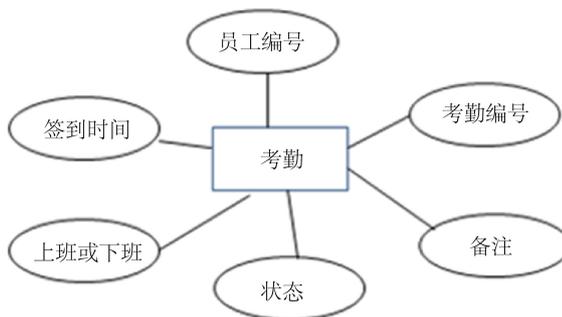


Figure 13. Attendance entity diagram
图 13. 考勤实体图

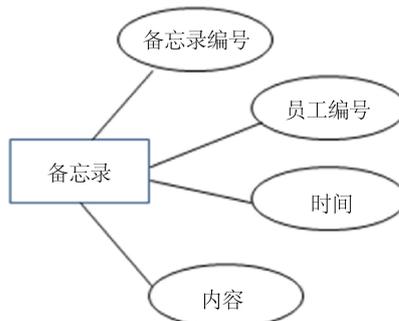


Figure 14. Entity Diagram of Memo
图 14. 备忘录实体图

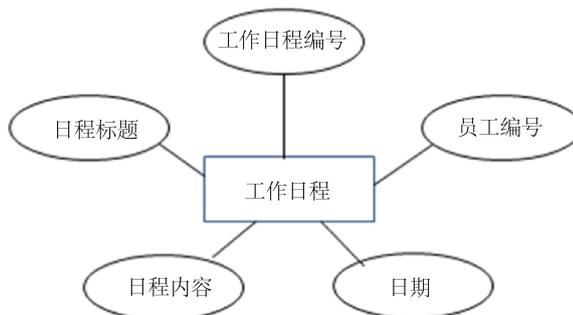


Figure 15. Entity diagram of work schedule
图 15. 工作日程实体图

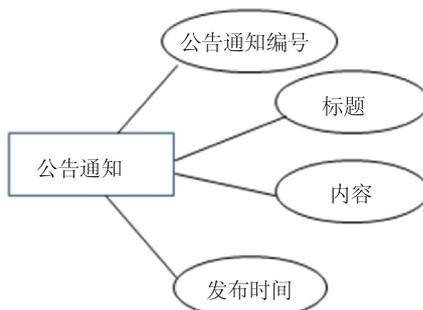


Figure 16. Announcement notification entity diagram

图 16. 公告通知实体图

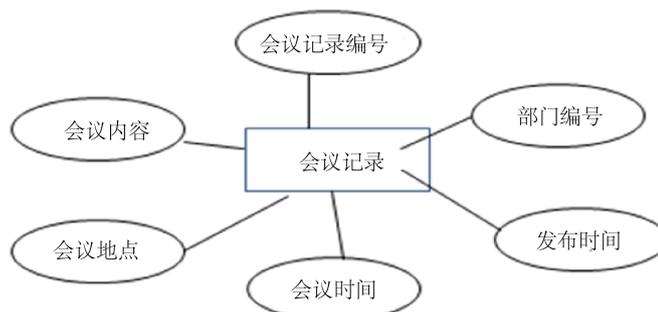


Figure 17. Entity diagram of meeting record

图 17. 会议记录实体图

5.2. 逻辑数据模型设计

将概念数据模型设计阶段的 E-R 图根据转换原则转换为逻辑数据模型设计, 在 OA 办公系统中主要涉及两种转换原则如下。

5.2.1. 三元联系类型的转换原则

此原则在数据库设计过程当中尤为常见, 在 OA 系统中, 根据分析, 部门、角色、员工之间的实体联系是 1:1:N 的关系。根据转换原则, 如果实体间联系是 1:1:N, 那么在 N 端实体类型转换成的关系模式中需要加入两个 1 端实体类型的键(作为外键)和联系类型的属性。

经过转换后的结果如下:

员工表(员工编号, 员工姓名, 角色编号, 部门编号, 联系电话, 邮箱, 状态);

部门表(部门编号, 部门名称, 部门级别, 部门经理, 上级部门编号);

角色表(角色编号, 角色名称, 权限组, 增加的权限, 修改的权限, 删除的权限, 查看的权限)。

5.2.2. 实体类型的转换原则

- 1) 需要严格执行将每个实体类型转换为一个关系模式的原则。
- 2) 对象实体的属性即为关系模式的属性。
- 3) 对象实体标识符即为关系模式的键。

经过转换后的结果如下:

菜单表(菜单编号, 菜单名称, 菜单地址, 菜单图标, 菜单类型, 父菜单编号);

考勤表(员工编号, 考勤编号, 签到时间, 上班或下班, 状态, 备注);

备忘录表(备忘录编号, 员工编号, 内容, 时间);

工作日程表(工作日程编号, 员工编号, 日程标题, 日程内容, 日期);

公告通知表(公告通知编号, 标题, 内容, 发布时间);

会议记录表(会议记录编号, 部门编号, 发布时间, 会议时间, 会议地点, 会议内容)。

数据库的设计和分析在面向对象分析时能够起到很好的深入作用, 使得在业务逻辑梳理的过程中更好的落地业务, 也能在结合实际案例进行分析时更深层次的理解面向对象分析方法, 理解其在系统设计技术中的主要应用, 对业务对象与对象之间的关系进行模型化, 进而有层次的逐步细化, 最终达到功能落地且符合实际需要的目的。

5.3. 系统研究展示

由于本文是对具有实际业务的 OA 系统进行面向对象的分析与设计, 因此对最终主页界面进行简单的展示, 同时也能够体现基于此系统的面向对象的方法的有效性和合理性, 同时能够更加直观的对中小企业 OA 系统业务需求进行简单的展示体现, 图 18 是系统的主页示例图。



Figure 18. System homepage diagram

图 18. 系统主页图

6. 总结

当前由于业务复杂程度的提高和系统体系的逐渐庞大, 传统的开发方法由于适应能力弱和难以检验正确性等特点, 已很难满足当前开发, 而面向对象方法强调以事务为中心, 相比传统的方法能够更加适合用来梳理分析和设计业务逻辑较为复杂的系统。因此本文主要以面向对象的思想和方法为研究, 结合 OA 系统的实际应用来更好的阐述和理解面向对象方法。

参考文献

- [1] 高素春. UML 在面向对象软件开发中的应用[J/OL]. 河南科技: 1-6. <https://kns-cnki-net.webvpn.cueb.edu.cn/kcms/detail/41.1081.T.20191213.1548.002.html>, 2021-04-23.
- [2] 管芳笛, 郭丽莹, 陈以君, 王红. 浅谈软件工程面向对象软件需求分析的研究[J]. 电脑编程技巧与维护, 2021(2): 22-23+54.

-
- [3] 田钟晓, 虞翔. 面向对象的软件工程中软件需求分析方法[J]. 电子技术与软件工程, 2017(16): 48.
- [4] 王智博, 周君贤. 办公自动化(OA)系统现状与存在的问题[J]. 信息与电脑(理论版), 2018(11): 127-128.
- [5] 钟卓霖. workflow技术在办公自动化系统中的运用[J]. 电子测试, 2021(4): 133-134+132.
- [6] 郝学柱, 董伟, 周爽. 企业 OA 办公自动化系统的应用与发展趋势[J]. 产业与科技论坛, 2017(2): 84-85.
- [7] 王晨, 刘鹏博, 冯双昌, 刘小畅. 协同办公自动化系统需求分析及设计[J]. 中国科技信息, 2021(Z1): 67-69.
- [8] 段冲, 孙芳. 关于企业协同办公系统的实施研究[J]. 黑龙江科技信息, 2016(34): 79.
- [9] Minhas, N.M., Masood, S., Petersen, K. and Nadeem, A. (2020) A Systematic Mapping of Test Case Generation Techniques Using UML Interaction Diagrams. *Journal of Software: Evolution and Process*, **32**, Article ID: e2235. <https://doi.org/10.1002/smr.2235>
- [10] Briggs, R.O. (2006) On Theory-Driven Design and Deployment of Collaboration Systems. *International Journal of Human-Computer Studies*, **64**, 573-582. <https://doi.org/10.1016/j.ijhcs.2006.02.003>
- [11] Yousefli, Z., Nasiri, F. and Moselhi, O. (2020) Maintenance Workflow Management in Hospitals: An Automated Multi-Agent Facility Management System. *Journal of Building Engineering*, **32**, Article ID: 101431. <https://doi.org/10.1016/j.jobbe.2020.101431>
- [12] Gaaloul, W., et al. (2020) Special Issue on Fog and Cloud Computing for Cooperative Information System Management: Challenges and Opportunities. *Future Generation Computer Systems*, **109**, 704-705. <https://doi.org/10.1016/j.future.2020.02.060>
- [13] Prasad, V.K. and Bhavsar, M.D. (2020) Monitoring IaaS Cloud for Healthcare Systems: Healthcare Information Management and Cloud Resources Utilization. *International Journal of E-Health and Medical Communications*, **11**, 54-70. <https://doi.org/10.4018/IJEHMC.2020070104>