**毕业设计说明书**

**基于DJANGO的“简颢科技公司”管理系统的设计与实现**

|  |  |
| --- | --- |
| **专 业** | **计算机科学与技术** |
| **学 生 姓 名** | **顾 子 健** |
| **班 级** | **B计算机172** |
| **学 号** | **1710704008** |
| **指 导 教 师** | **李 运 平** |
| **完 成 日 期** | **2021年 6月11日** |

基于DJANGO的“简颢科技公司”管理系统的设计与实现

**摘 要：**随着人工智能和大数据的步履逐步深入、企业的发展与转型不断地革新，企业信息管理也面临着多样化，层次化，信息化的多向机遇。将信息以数据化，智能化，可预测的方式呈现出来即为时代所趋。

系统主要采用了Python Web Django框架，根据MVC模式设计改变的新型结构模式MTV，同时前端运用了jQuery和Bootstrap等框架，同时使用了部分Vue来渲染全体元素布局。

简颢科技管理系统主要有四大优势。第一是公司上级的通知显示及人员信息更新。第二是对公司的收入与支出的实时监控。第三是良好的市场数据处理，及未来数据推演。第四是数据收入支出的分类占比可视化，以便公司在未来及时地对资源进行分配与调度。

通过简颢科技管理系统，公司内部员工的信息及数据可以及时更新并实现可视化，操作方便快捷，在大大提高了公司对人员数据的整合能力上，并能更好地让上级了解当前公司的收入及支出占比。系统是针对原先简颢科技公司管理软件的一次更新，摆脱原先的管理束缚，而应时代更新和发展推出的一种企业信息管理的升级产物。

关键词：EIM；Django；可视化数据

Design and Implementation of JH Technology Management System Based on Django

**Abstract:** With the gradual deepening of artificial intelligence and big data, and the continuous innovation of enterprise development and transformation, enterprise information management is also facing the multi-directional opportunities of diversification, hierarchy and informatization. Therefore, it is the trend of the times to present information in the form of data, intelligence and predictability.

The system mainly adopts Python web Django framework, and designs a new structure mode MTV according to MVC mode. At the same time, the front-end uses jQuery and bootstrap framework, and uses some parts of Vue.js to render the layout of all elements.

JH technology management system has four advantages. The first is the notification display and personnel information update of the company's superior. The second is the real-time monitoring of the company's income and expenditure. The third is good market data processing and future data deduction. The fourth is the visualization of the classification proportion of data revenue and expenditure, so that the company can allocate and schedule resources in time in the future.

Through JH technology management system, the company's internal staff information can be updated in time, data visualization, convenient operation, timely data update, and other aspects of optimization, which not only greatly improves the company's personnel data integration, but also better let the superior understand the current company's revenue and expenditure proportion. To sum up, this system is an upgrade product of enterprise information management, which is aimed at the original management software update of JH technology company, getting rid of the original management constraints, and in response to the update and development of the times.

**Keywords:** EIM; Django; Visualization data

目 录

[1 概述 1](#_Toc5605)

[1.1 简颢科技的管理现状 1](#_Toc28333)

[1.2 解决方案论证 2](#_Toc12000)

[1.3 相关技术 2](#_Toc1303)

[2 系统配置 3](#_Toc6835)

[2.1 技术前提 3](#_Toc15114)

[2.2 设备参数 3](#_Toc30900)

[2.3 开发工具 3](#_Toc23832)

[3 系统分析 3](#_Toc11128)

[3.1 可行性研究 3](#_Toc4770)

[3.2 需求分析 4](#_Toc22388)

[3.3 系统目标 9](#_Toc1662)

[4 系统设计 10](#_Toc31635)

[4.1 概要设计 10](#_Toc26026)

[4.2 模块设计 11](#_Toc7057)

[4.3 详细设计 15](#_Toc27846)

[5 数据库设计 17](#_Toc29212)

[5.1 概念结构设计 17](#_Toc16937)

[5.2 逻辑结构设计 19](#_Toc4360)

[6 系统实现 21](#_Toc2505)

[6.1 登录入口功能的实现 21](#_Toc18458)

[6.2 人员数据收集功能实现 22](#_Toc18777)

[6.3 公司财务部分功能实现 23](#_Toc12087)

[6.4 市场模块功能实现 26](#_Toc9790)

[6.5 数据分析及可视化模块实现 27](#_Toc22262)

[6.6 消息通知功能实现 29](#_Toc5200)

[7 系统调试 29](#_Toc31470)

[7.1 首页登录界面的调试 29](#_Toc897)

[7.2 人员数据收集的调试 31](#_Toc11056)

[7.3 公司财务功能调试 32](#_Toc19423)

[7.4 市场功能的调试 33](#_Toc20170)

[7.5 数据分析及可视化功能的调试 35](#_Toc21139)

[7.6 通知功能的调试 35](#_Toc12113)

[8 结束语 37](#_Toc30651)

[参考文献 38](#_Toc32190)

[致 谢 39](#_Toc7265)

[附 录 40](#_Toc801)

[附录1 源程序清单 40](#_Toc18650)

[附录1.1 项目前端主要模块文件（依赖注入） 40](#_Toc1011)

[附录1.2 项目后端处理文件（所有处理内容，不包含跨系统数据库内容） 40](#_Toc32585)

[附录1.3 项目MyJsList.js文件（三种重要算法） 57](#_Toc21273)

[附录1.4 项目页面文件 58](#_Toc25376)

基于DJANGO的“简颢科技公司”管理系统的设计与实现

1. 概述

该系统是基于Django的“简颢科技公司”管理系统。在这个人工智能和大数据发展极度火热的社会环境下，企业的信息化管理的趋势不再是单纯的数据整合，而是通过数据实现可视可观可查可预演来对企业的动态管理。而这也顺应了企业信息化管理的时代二次更替，来提高简颢内部的信息的管理效率。

在原先数据化的管理下，我们只会通过在数据库中调用数据来实现对数据的了解，而缺少对其的管理和利用。同时，通知的传递，通过现在主流的聊天工具，很难做到信息加密和隐秘传达。伴随数据的庞大与公司人员数量的增加，数据的可视化、人员信息的个体化、通知的私有化，在简颢科技管理系统中都能得到很好地展现。

系统涵盖功能不仅能降低原先通过Excel进行人员数据的低效性，还能通过公司收入和公司支出来评判公司的实际状况，更能通过对大数据的预演，实现实时观测，这才能合理地调动公司资源，来对公司内部某一块进行实时的调整更新。

1.1 简颢科技的管理现状

**1.1.1 信息管理的方法过于落后**

简颢科技企业现在仍然采用传统的人员数据收集和公司数据库直调的方法，这会造成一些问题。首先人力数据收集的成本加大，且耗费时间较长。当要收集人员数据收集，或是想通过数据库数据查看公司近况时，效率是极其低下的。其次，纸质化办公或是原先的Excel数据收集，在大量数据的归类规整方面，会浪费大量的人力。其中还不乏因内部员工数据未申报导致信息不全的状况。

**1.1.2 信息管理的展示不够明晰**

简颢科技企业对于信息化管理原先一直保持着数据调用，会议分享的方式。但是单纯的ppt演示和从嘴中得来的数据，很难对数据有一个直观的感受，有时对于新员工或者新部门来说，可能不太能知道各个数据之间的差距。

**1.1.3 信息管理的流程太过繁琐**

在原先的信息管理系统中，一直都使用的Excel数据直接导入数据库的方法，或是通过软件进行简单的数据整合。数据存储在数据库中，很难对大量数据进行查看，调出一些数据的时候，不会对数据归类，数据一盘散沙，分隔开来比较困难。

**1.1.4 信息管理的目的不确定**

很多管理层不知道其他部门的效益数据，不知道自己部门在公司中的贡献程度，很难做出下一步应该怎么调整部门的走向。

1.2 解决方案论证

“简颢科技公司”管理系统采用的是B/S基础架构，在使用和数据整合方面简单快捷，可以做到对原先系统的革新。

在针对信息管理手段落后的问题上本系统能做到从传统企业数据管理到新可视信息化的转型，通过信息化数据管理手段将数据整合成包在通过分类来解决信息归类繁琐等一系列问题。其中，涉及的模块有人员数据收集功能，公司收入支出数据收集及可视化展示功能，市场产品收入数据汇总及公司内部通知。

而信息化管理不仅具备了数据收集，对大量数据的分筛，数据清洗等都做了优化，让数据操控和显示更加清晰明了，并能通知模块实现消息私有化、加密化。

1.3 相关技术

a) Django：Python下有很多不一样的 Web 框架。Django是所有选手中最有代表性也是最能承接时代Web变革的一位。Django 最初设计用于快速开发新闻类站点、快速生成类站点，目的就是要实现网站开发的快捷高速化。Django 采用了 MVT 的软件设计模式，即模型（Model），视图（View）和模板（Template）。对原先MVC结构实现了自己的定义。

b) jQuery+BootStrap：jQuery框架和BootStrap框架减少了设计前端麻烦的同时，可以对页面的布局和样式施以控制，以达到更加惊艳的效果，比如说jQuery中的Animate动画组件和BootStrap中的Carousel轮播组件。Animate对原生JS中生成动画的代码进行了优化，在原生JS中，使用计时器的方式实现的动画，其代码较为复杂，而Animate组件更能动态和渐进的展示动画效果。而Carousel组件是Bootstrap自主集成的组件，原生JS的代码中实现所需要的左右图片切换功能是十分繁琐的，而Carousel可以直接实现相应功能。

c) Numpy+Pandas+BigData：Numpy一个强大的N维数组对象，支持大量的数据运算及处理，集成C / C++和Fortran代码的工具，这也是众多机器学习框架的基础库。Pandas是python最常用的库之一，其主要用于数据分析处理，如数据切片、数据清洗等。大数据（big data），指无法在一定时间以内，使用常规办法进行捕捉、管理和处理的数据流集，而需要新的数据处理模式才能更好地对海量、高增长率和多样化的信息资产实现调控。

d) Linear Regression+Python：借助Python数学模块实现的线性回归分析，实现对未来数据的呈现。Linear Regression线性回归：线性回归使用最佳的拟合直线（也就是回归线）在因变量（Y）和一个或多个自变量（X）之间建立一种关系。回归分析（regression analysis）指的是确定两种或两种以上变量间相互依赖的定量关系的一种统计分析方法。

1. 系统配置

2.1 技术前提

在使用相关技术之前，是要具备一定的编码知识，如JS，Html+Html5，Css+Css3+部分Scss，Python，MySQL语句。JS，Css和Html为前端的页面提供了基础的页面设计及脚本布局，在掌握基本编码语句的情况下，可以转而使用jQuery框架和BootStrap框架来提升整个页面的框架集成度和美感，简化编码内容的同时，提升你的页面在制作出成品后能展现的实际效果。Python语言是具有跨时代意义的新兴编程语言及脚本语言，而BigData、Numpy及pandas模块的使用都建立于会编写Python代码的基础上，是一种提升系统性能和完备系统功能的最佳选择，通过这两个模块提供的功能能减轻系统基层数据处理全部功能的工作量，易于后期更新维护。Python Math+Linear Regression是在基础数学库Math的基础上进一步提升的数学模块，它们是针对收集到的数据进行二次排列整合的最佳工具，所以必须熟悉掌握MySql的基本语法及数据取出，如何通过方法实现的基本逻辑及原理。

2.2 设备参数

a) 计算机平台：X64

b) 操作系统：Windows 10

c) 运行环境：Python Web Frame+MySQL+Django

d) 内存：16GB

e) 硬盘：500GB或更大

2.3 开发工具

a) PyCharm

b) MySQL Workbench 8

c) python 3.9.2

d) Django 3.1.7

d) Google Chrome

1. 系统分析

3.1 可行性研究

**3.1.1 技术可行性**

科技公司在信息管理方面大都采用效率较慢且高成本低回报的传统消息管理机制。针对简颢科技管理工作，在原先管理软件的基础上部署了全新的信息化革新模块，有效地弥补传统信息化管理办法带来的不足及缺失，同时将信息化管理模式标准化。这样整个系统所使用的Django框架功能，就能保证在原先系统的稳定下保持新系统的迭代，根据原有数据实现可视化信息化智能化的管理。

**3.1.2 经济可行性**

经济可行性关注于成本是否在可控范围内，同时在技术成本上性价比是否如期。但这恰恰是传统信息化管理的两大致命点，反之，也是“简颢科技公司”管理系统的两大闪光点。通过在原先系统沿用的同时，兼容数据库操作，分析数据，规整数据。节约成本的同时，还为信息化管理增添了可视化可预测的全新技术支持。

**3.1.3 操作可行性**

这套系统是原先繁琐设计的革新，原本繁琐的数据库数据存入收集，都在本系统中得到的改善。企业最担心的是原有系统已经很成熟，新系统是否会不兼容原有的内部操作，但本系统完美地去除了原有系统的糟粕，而简单快捷的设计，更加方便后期的使用。

**3.1.4 可行性研究结论**

综上所述，“简颢科技公司”管理系统是一个具有可拓展性的系统。不仅成本可控，新老系统分离，操作简便，最主要的是符合公司的信息操作期望趋势。企业优化在未来的部署及规划上，一定是格外关注信息可视，信息推演，数据集成，效率优良这一块的，而“简颢科技公司”管理系统恰恰作用于解决传统信息管理方法带来的不快捷不方便与不明晰。

3.2 需求分析

**3.2.1 需求描述**

目前很多企业都想拥有属于自己的交流渠道和通知渠道，毕竟公司的消息和通知通过常用聊天软件很难做到加密，及时通过加密手段，改变原文传输方式，实现可靠传输。在数据生成或是Excel数据收集时，成千上万条数据，多少会出现效率低下，表现力不足的情况。针对现公司通知模块和支出收入模块，本系统更偏向于数据分类、统计和预演的操作。

在“简颢科技公司”管理系统中，超管（SA）是权限最大的角色，有权操作系统的所有功能，普通管理员（Admin）能实现通知的发送，用户的权限由系统默认分配，成为普通用户权限（user）。管理员能对所有收集的数据进行增删改查以及通知模块的消息发送。当然超管的权限是包含所有管理员权限的。普通用户是可查看日常通知，收入支出由不同部门人员发送，即普通用户都就有的发布权限，但只能发布和更新及查看最新管理员消息的权限。普通用户操作流程如图3-1。同时，我们将普通员工操作分为市场部人员操作和其他部门人员（普通人员）。由于市场部人员负责产品的收入模块，所以相对于普通用户独立。市场部门用户操作流程如图3-2。

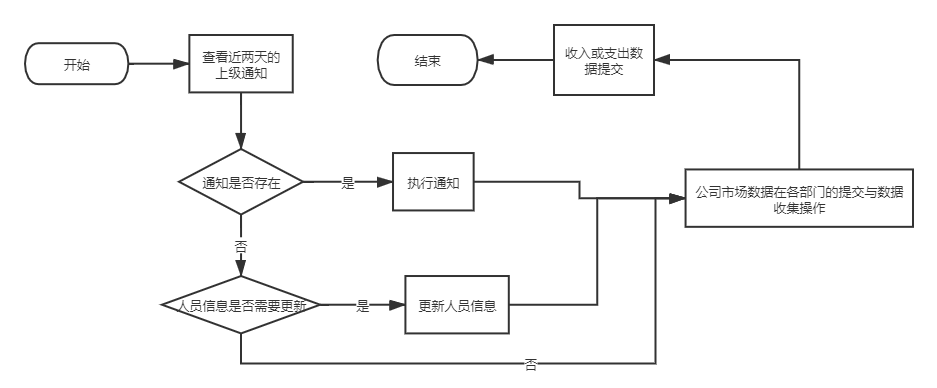


图3-1 普通用户操作流程图

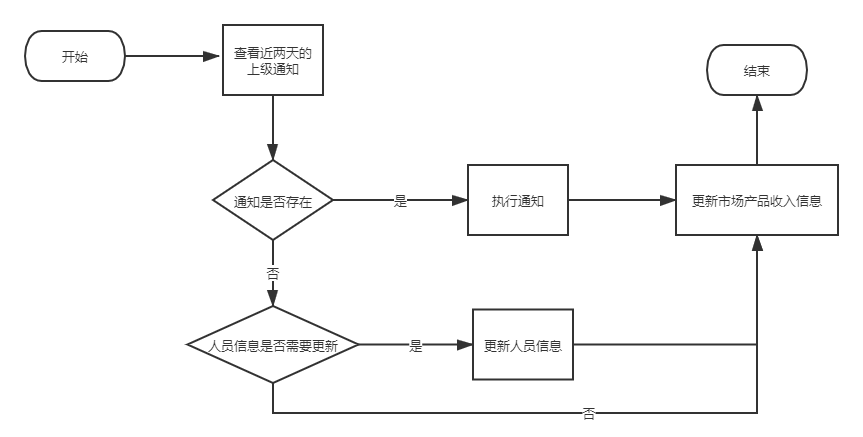


图3-2 市场部门用户操作流程图

由于普通员工需要得到管理员的通知后执行相应操作，具有强依赖性。所以，我们在设计时将管理员的操作，分成了全时段，而非人员等待通知后执行，而全时段生命周期设置为两天。故管理员需要每日下达近两天内的指令和通知，以方便每个部门的数据汇总和统计及时。在原先公司系统中每日的市场消息是具有时效性的，所以实际的市场数据，在我们获取到未经处理的数据之后，可以通过市场部发布的最新收入消息得到新数据。同时管理员需要通过图表绘制演示和未来推演分配后面的计划和任务。管理员操作流程图如图3-3。

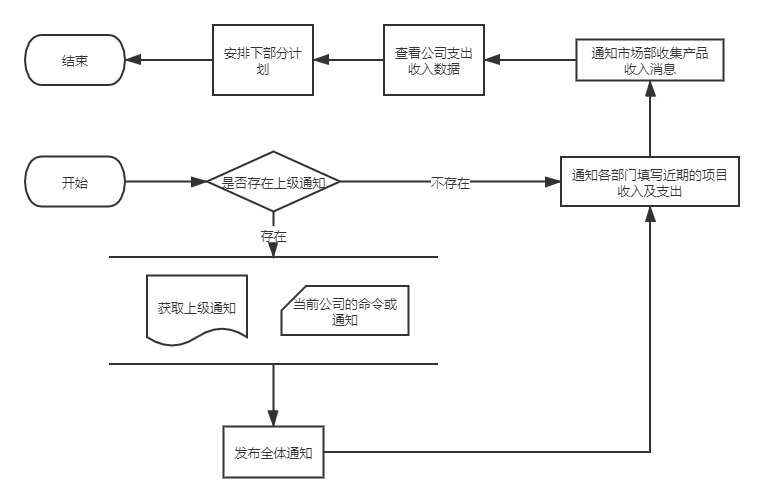


图3-3 管理员操作流程图

当然根据员工的职位不同，所需要负责发布的内容同样也是不同的。比如数据库管理员或是其他负责公司额外收入支出的人员可以对数据库实行修改，而大部分公司员工是无权对其操作的。

**3.2.2 用例建模**

a) 管理员：首先系统管理员的权限是最高的，系统中所有的功能均可以使用。但是系统管理员主要负责的部分还是通过获取系统生成的信息来部署下一步计划和通知的发布。在系统信息管理模块中，管理员是作用于第二层的，而第一层的超级管理员负责软件的维护和数据库的实际操作，当然管理员的作用几乎和超级管理员一样，只不过当系统运行出现问题和系统数据出现错误或失效时，超级管理员可以直接对数据库执行操作，其作用类似于DBA，当然如果出现收入支出信息填写出错，可以通过更新来更替掉原先的数据。同时，超级管理员，可以实施对数据库的增删改查操作，实现新员工账号添加，由于系统是内部信息管理系统，账号是由超级管理员统一安排发放，并且管理员分配也由超级管理员部署。详见“简颢科技公司”超级管理员的用例如图3-4。

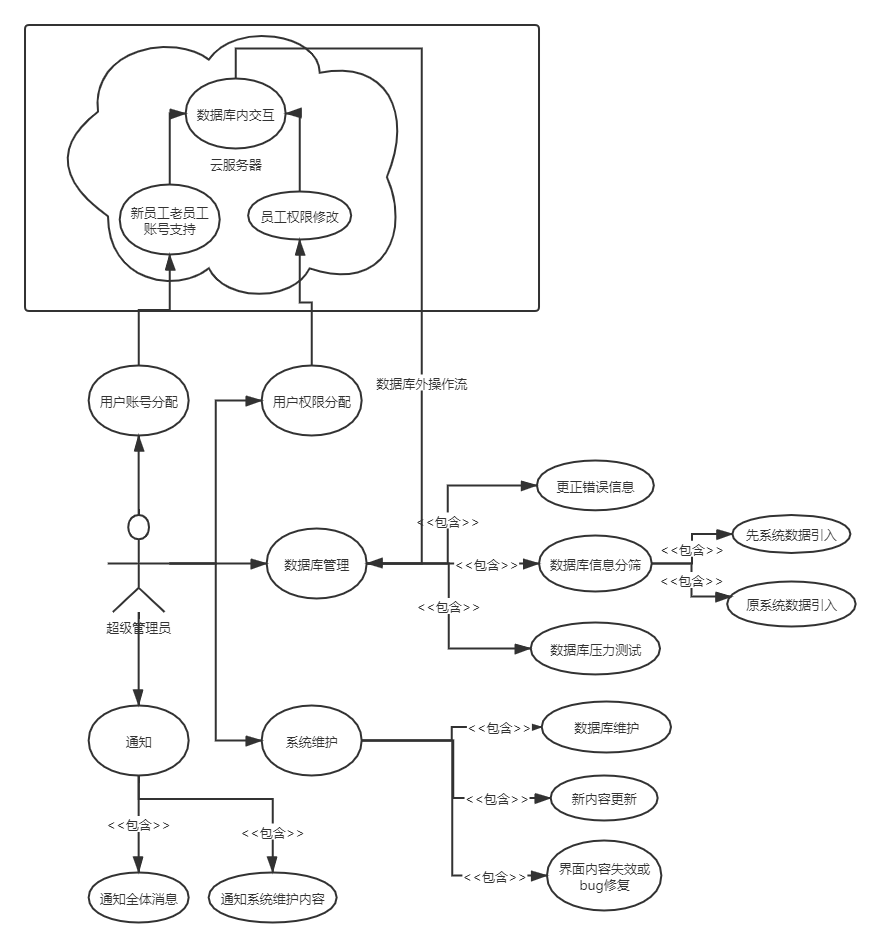


图3-4 简颢科技管理系统超级管理员的用例图

b) 管理员：管理员登录后可以执行页面的通知，信息获取，收入支出信息的更新或是通过获取到的信息来分配任务，对数据实现预演。由于本系统实现的是数据公开化，所有员工均可开到公司的内部收入支出，当然，只能看见归类的信息，其它类型的支出和收入信息，只存在于数据库，管理员可以同超级管理员一同查看详细项目内容。详见简颢科技管理系统管理员的用例如图3-5。

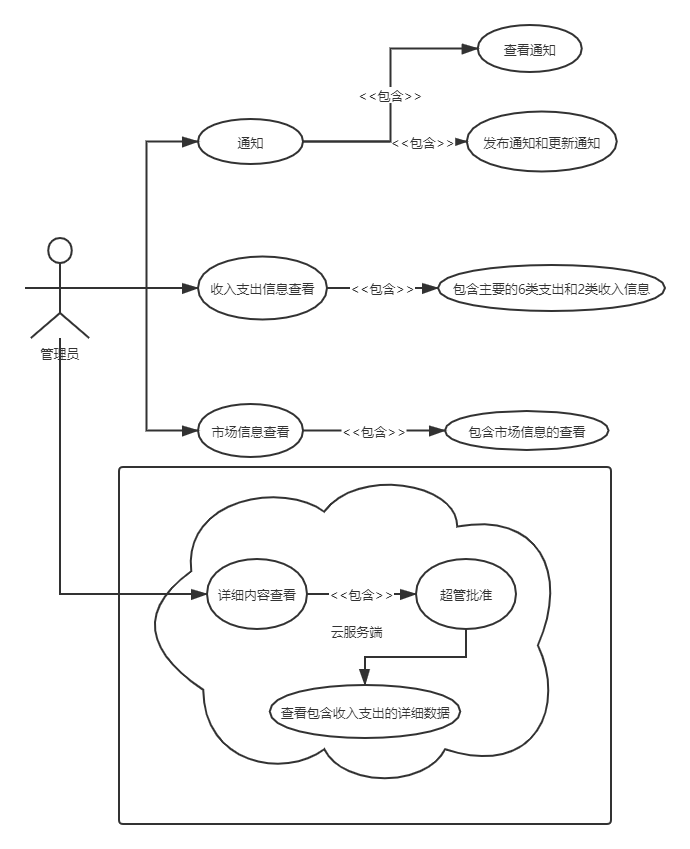


图3-5 简颢科技管理系统管理员的用例图

c) 普通员工：普通员工可以使用通知模块的信息查看，查看到近2天的消息通知。但是，普通员工不可实行通知的发送。普通员工可以通过发送部门收集到的数据来更新同步当日的收入支出数据。同时，这些数据的发送都会被登记，并存入数据库中，这样很大一部分可以知道是否有恶意填写数据或恶意更新数据的可能。当然，本系统也给予了普通员工查看公司数据的权限，可以查看到主要的市场收入来源和主要的公司支出和收入，详细信息除外。详见简颢科技管理系统普通员工的用例如图3-6。

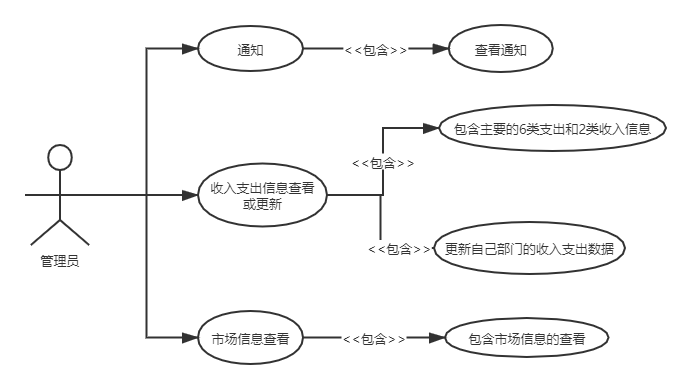


图3-6 简颢科技管理系统普通员工的用例图

3.3 系统目标

企业EMI管理系统是建立在信息技术基础上，以系统信息化管理的思想，为企业决策层及员工提供快捷不繁琐、透明的信息管理平台。

在简颢科技管理系统中，我们设定了不一样的数据管理技术目标，脱离原先的枯燥数据，不能直观地查看数据的苦恼。以MySql做基底，建立了新数据库与原数据库的数据进行综合分筛内部管理，这样脱离系统的内部操作，减去了大量人力。建立了信息资源共享机制，实现了对数据的授权访问控制的机制。同样安全也是我们考虑在内的，实现了系统识别与应用的安全可靠运行，支持了csrf跨域攻击反制，XXS漏洞攻击反制，SQL注入等安全技术的加持。对每一个项目或者每一笔支出和收入的归类分析，对数据的追踪处理，及时更新老数据，让数据保持时效能力。

我们也为系统设计了三种不同的角色，并给他们赋予了各自的特色功能。超级管理员在实施对数据库的增删改查操作的同时，可以实现新员工账号添加，权限的划分。本系统内部信息管理，其账号是由SA统一安排发放的，并且管理员权限的分配也由SA全权设置。管理员登录后可以执行通知功能的操作，各收入支出信息的获取，收入支出信息的更新或是通过获取到的新信息来分配实际操作指令的通知，对数据表图执行重新绘制。普通员工可以使用通知模块的信息查看功能，查看到近二天的管理员即上级通知。但是，普通员工不可以使用通知的发送功能。普通员工可以通过收集到的本部门数据来更新近二日的收入支出项目，以完成上级通知的内容。数据公开透明化，数据可视化，数据可预演化，是本系统最终的目标。这样也能很好地提高公司内部之间的协同性，凝聚性，领导的及时督查程度和信息的可视共享性。

1. 系统设计

4.1 概要设计

简颢科技管理系统的功能模块有七大主模块，主要有首页，快捷菜单模块，人员数据收集模块，公司财产事务模块，市场盈亏数据模块，数据分析及可视化模块，内部消息通知模块七大模块组成，这些大模块中可能包含小模块，由于本系统是承接原系统实现的新系统，只迭代了原先的部分功能，使原有系统中的数据依旧发挥作用。

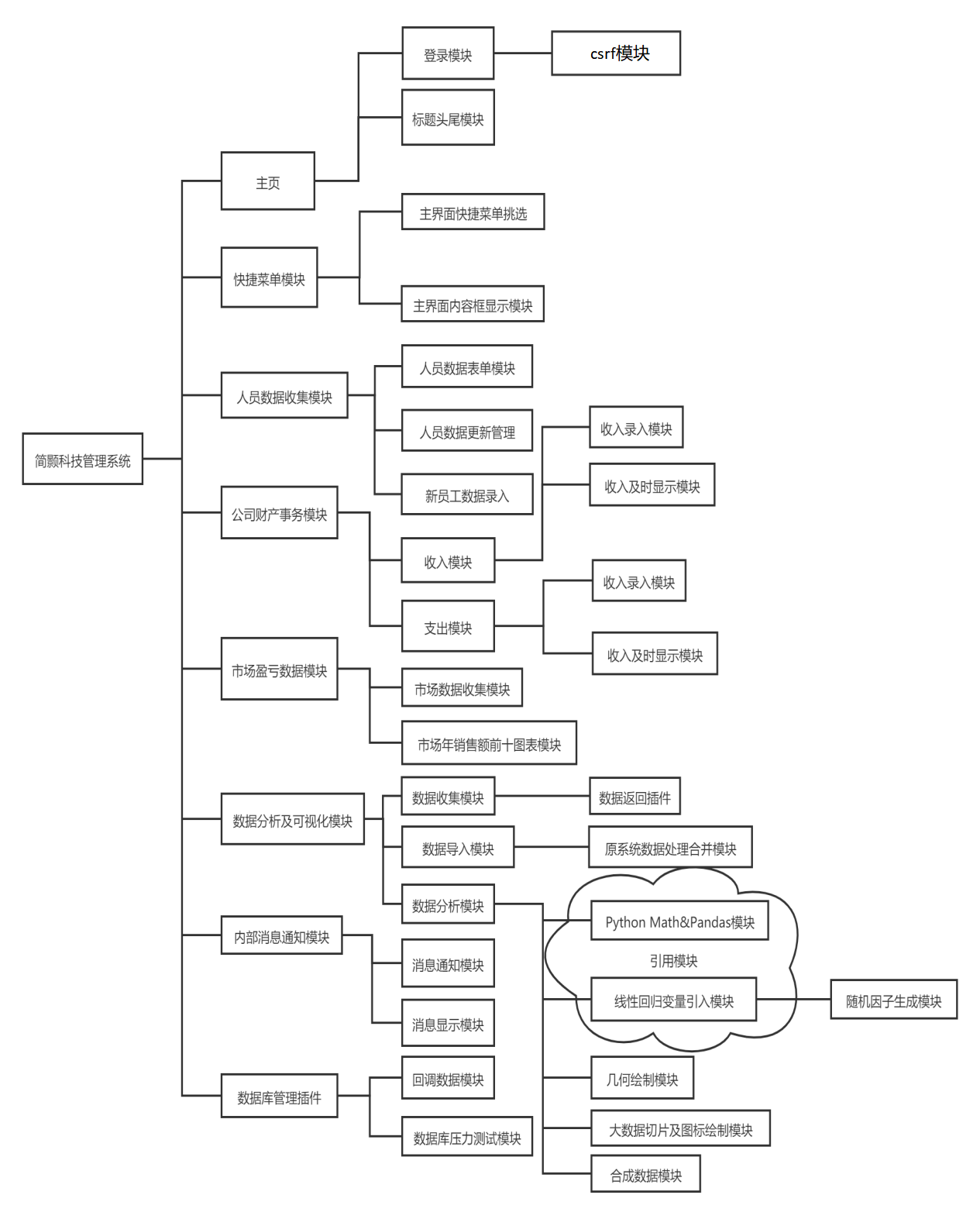


图4-1 简颢科技管理系统的功能框架图

4.2 模块设计

a) 首页：其包含登录模块和标题头尾模块，其核心部分还是登录模块，设计初衷是准备放入注册模块的。但是，考虑到公司内部集成化的缘故，只设计了登录模块，并赋予登录模块csrf防跨域攻击模块，保证外来人员或其他局域网人员对系统造成威胁。其流程详见登录模块主流程如图4-2。

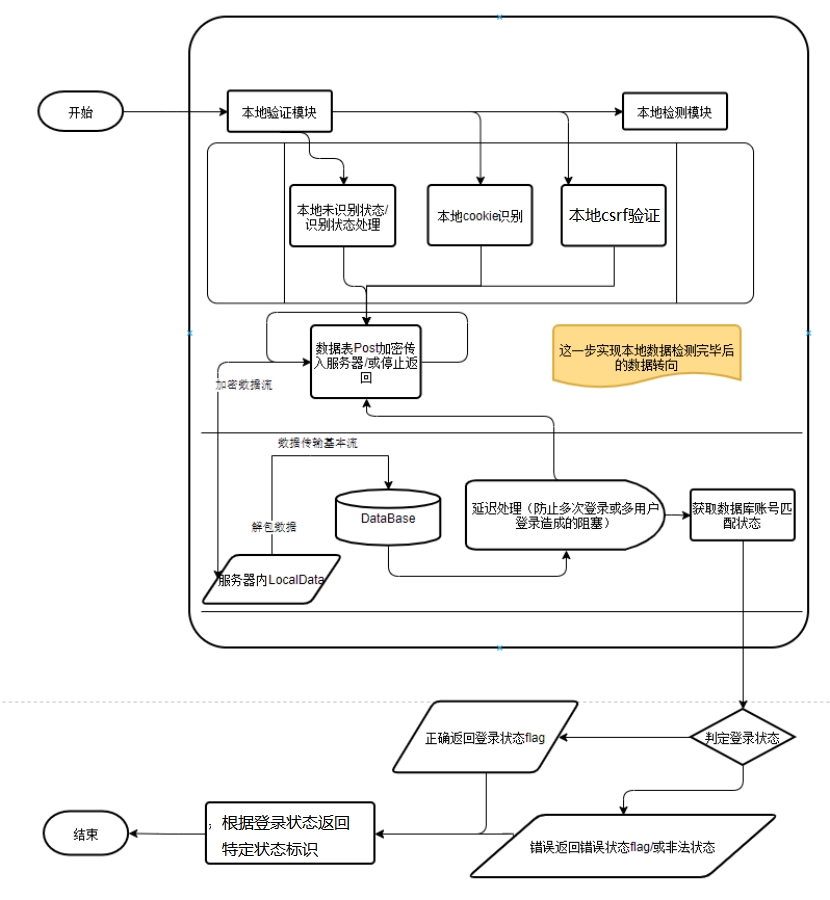


图4-2 登录模块主流程图

b) 快捷菜单模块：登录后主界面的主要选项卡模块，实现本系统的所有模块跳转和刷新，其主要实现跳转功能。同时，其具备对实际对象的判定功能，实现跳转到某个功能实现某个模块下的特定时钟事件的作用。当鼠标移动到任何一个选项卡上，内容框部分会对应显示子功能。其流程详见主界面快捷菜单动作流程如图4-3。

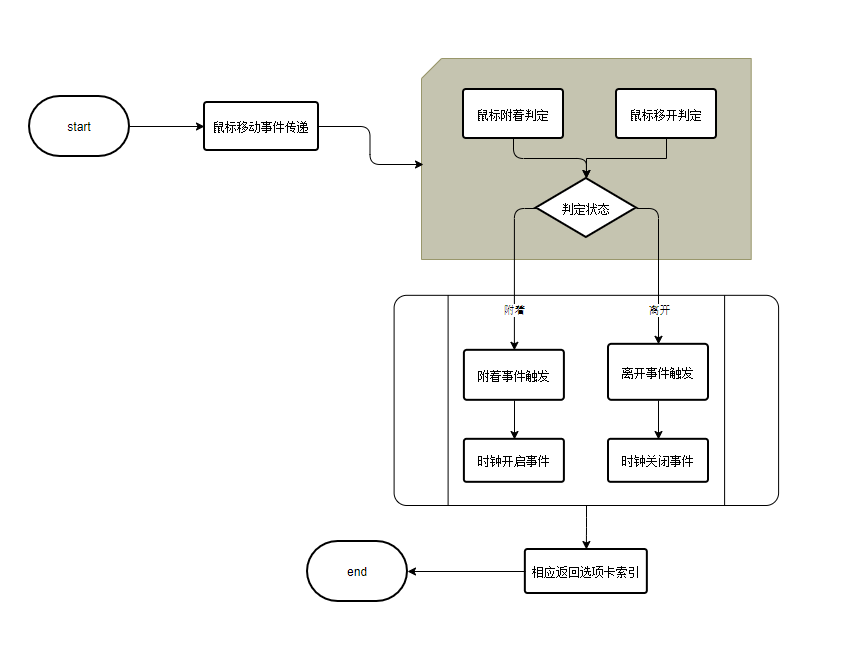


图4-3 主界面快捷菜单动作流程图

c) 人员数据收集模块：在此模块中，超管在数据库中可以收集到来自用户们填写的数据信息，并实行定时提醒更新通知。如果是新用户，新收集的数据会新增到我们的数据库中，如果是老员工，我们会更新他本次填写的新内容。其流程详见人员数据收集模块流程如图4-4。

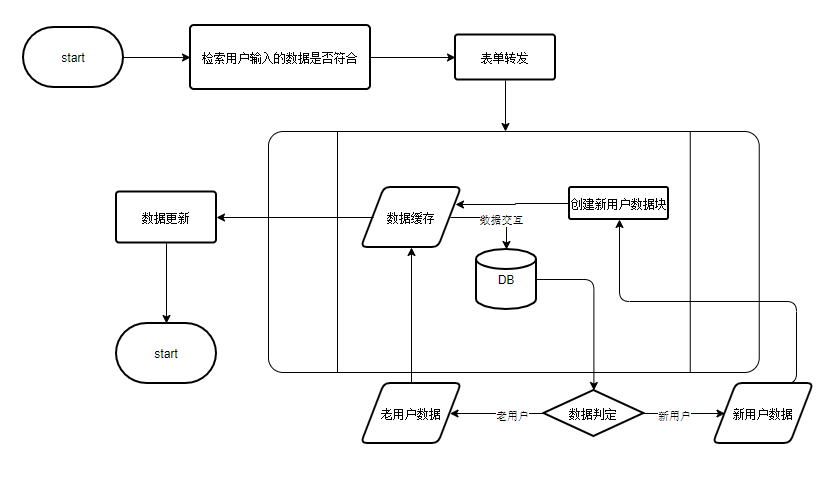


图4-4 人员数据收集模块流程图

d) 公司财产事务模块：在公司财产事务模块中，我们使用了AJAX动态事件和分布处理流，实现了两个线程的不同操作。同时，收入支出的信息录入也是必不可少的，其流程详见收入支出更新流程如图4-5。

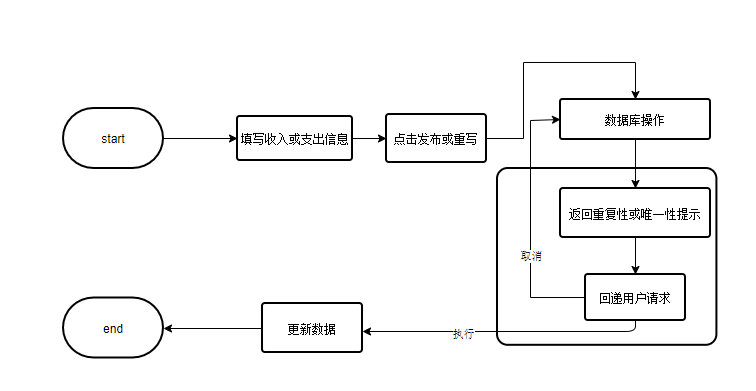


图4-5收入支出更新流程图

下面是对数据显示的功能流程图。首先，对数据库中的数据进行检索，并且将此数据从后端返回至前端，局部刷新以实现我们的动态事件部分。其次，分布处理和双线程部分，主要实现的是支出和收入模块的相互隔离，使原先的数据能够被不同子程序流处理，返回动态的数值，也是节约系统资源的一种办法。详见收入支出模块流程如图4-6。

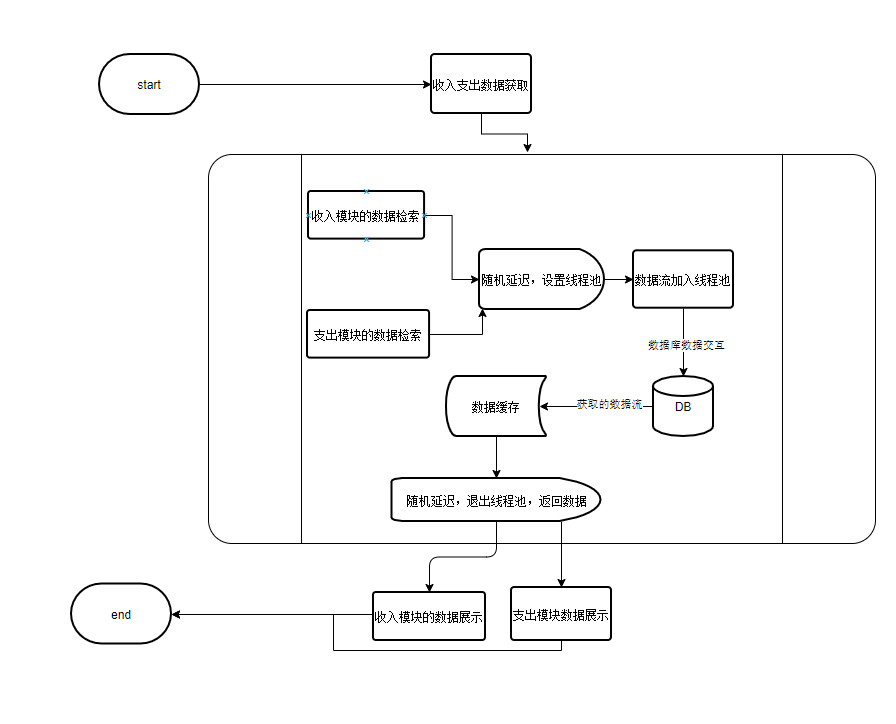


图4-6收入支出模块流程图

e) 市场盈亏模块：在本模块中，我们实现的是对我司市场产品的信息录入，并分析得出本年中最高销售额的十个产品。其主要用于分筛原系统数据并清洗数据来获得新数据，并实现图表的绘制。其流程详见市场产品信息交互流程如图4-7。

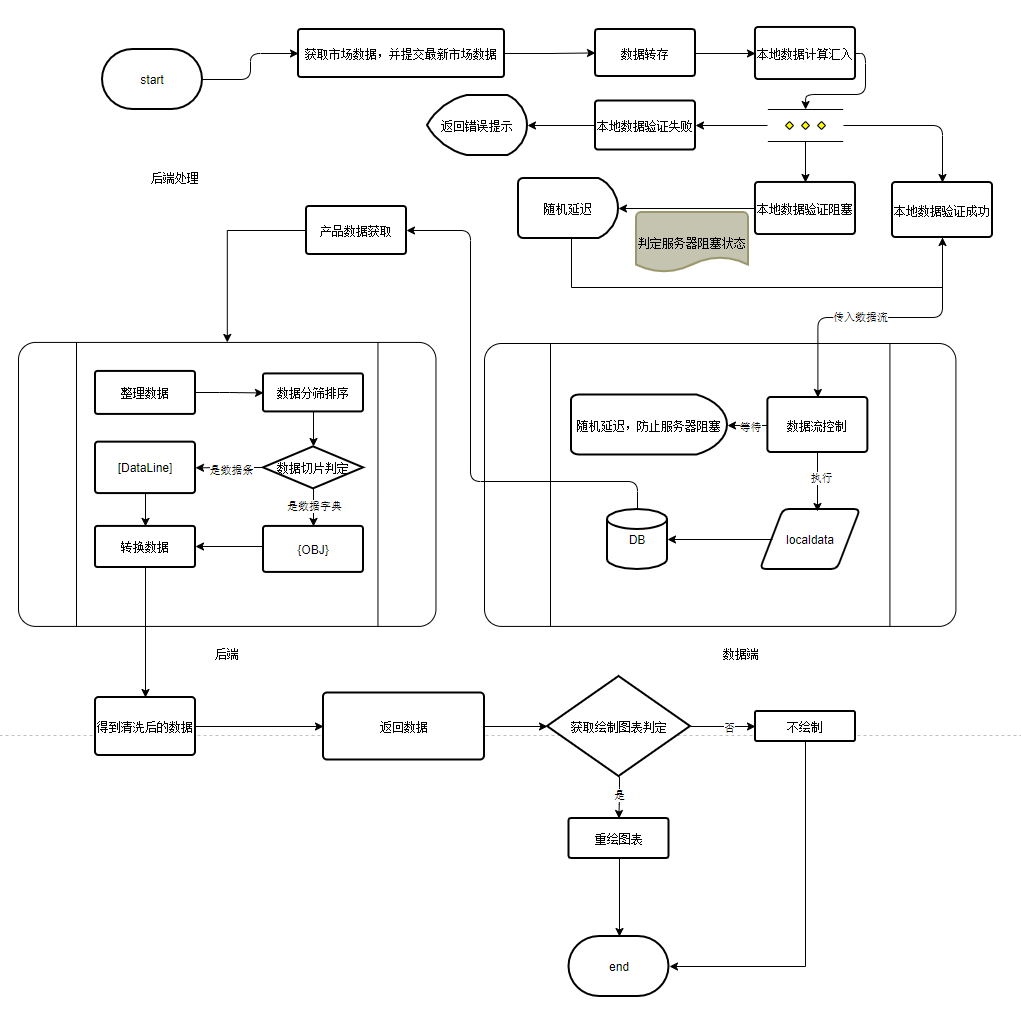


图4-7 市场产品信息交互流程图

f) 数据分析及可视化模块：这个模块也是整个系统中的精髓所在，通过获取公司收入支出模块的数据和公司市场产品的数据来进行数据分析。并画出数据占比分析图，通过时间因子，即随机因子来确定线性方程，并绘制预测函数的图表。而本模块更注重用户对数据的可视体验，所以具体获取数据和操作的流程，会在详细设计中展示。

g) 内部消息通知模块：在内部消息通知模块中，我们主要实现的是传达上级通知和用户接收消息的体验。其流程详见内部消息通知流程如图4-8。

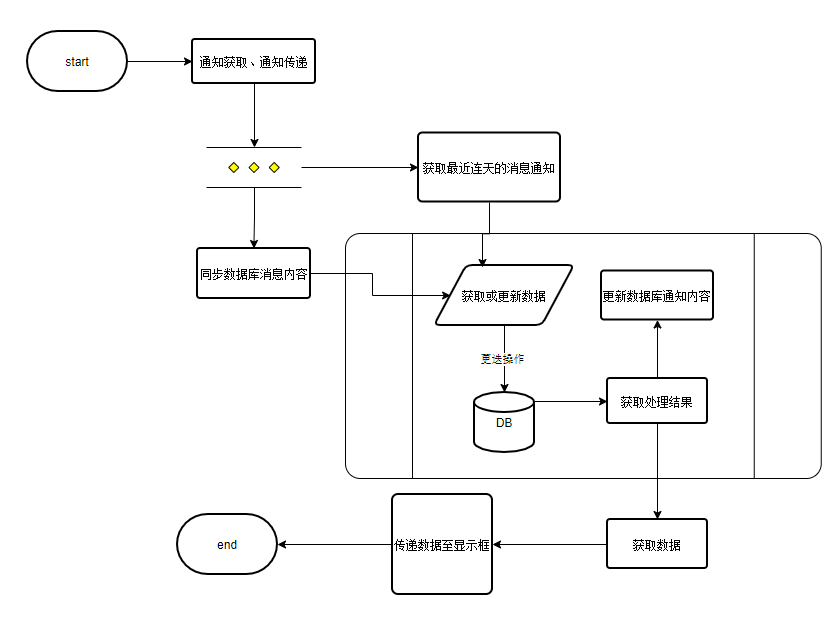


图4-8 内部消息通知流程图

4.3 详细设计

本系统中每个不同的模块功能实现的子功能都是不同，在个别功能上会对用户权限实现部分限制，对于不同的角色的不同作用或者说我们需要不同的用户使用其不同功能以实现不同的作用，本系统在原简颢科技系统的基架上搭设，功能不包含原先已经实现的部分，是一次系统的升级。下面是对不同用户使用所做的说明。

a) 管理员：在简颢科技管理系统中，管理员的权限覆盖了所有系统所涉及的功能模块。但是系统管理员的操作基本都是对数据库的操作，包含对人员权限的设置、用户的账号密码更新，以及部分数据的更迭。当然，SA的作用包含其中所有但不仅限于此，额外添加了数据库压力测试，系统维护等。

b) 普通用户：普通用户分为市场部和其他部门人员，用户每天登录需要先查看当日的最新消息和最新通知以获得今天要完成的任务，如果没有最新的任务，一般用户们需要提交本部门所设计模块的支出和收入模块的录入，当然数值的异常，系统会在每天SA检验后，再实施更改，或通知更改。用户会在第二天收到消息。市场部人员需要对市场产品信息进行更新，当然数据也会经过SA检验。普通用户的日常操作基本如上所述。

管理员每天会发布新一天的指令以方便员工们查看，在此期间通知模块可能会存在管理消息未发送而员工提前查看的情况，不过这个情况我们一般不必担心。因为最新的消息会在第二天仍然被转发，同时一个部门的消息是在不同员工登录时更新的，员工也可以互相转告，以达到相同的效果。所以管理员和用户的访问通知的状态有两种。详细可见通知流程图4-9。

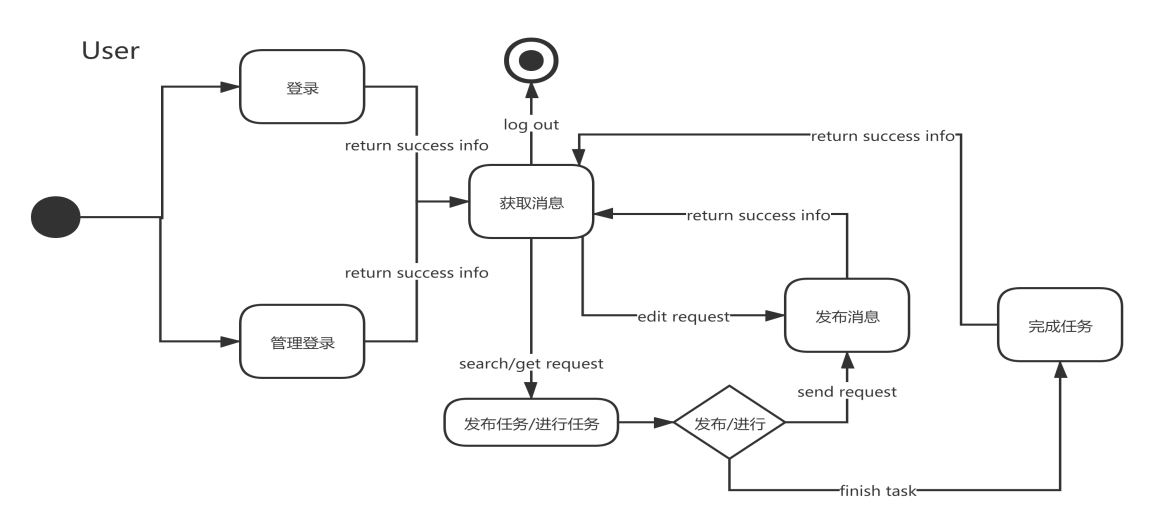


图4-9 通知流程图

接下来就是每一个角色执行的活动行径。首先，用户在没有收到新消息时，应该执行什么，在接收到新消息后应该怎么做。其次，我们的管理员每天发送消息和新通知的一个活动路线，以及何时查看图表最佳，即信息的时效性。详细活动图可见不同角色操作活动图4-10。

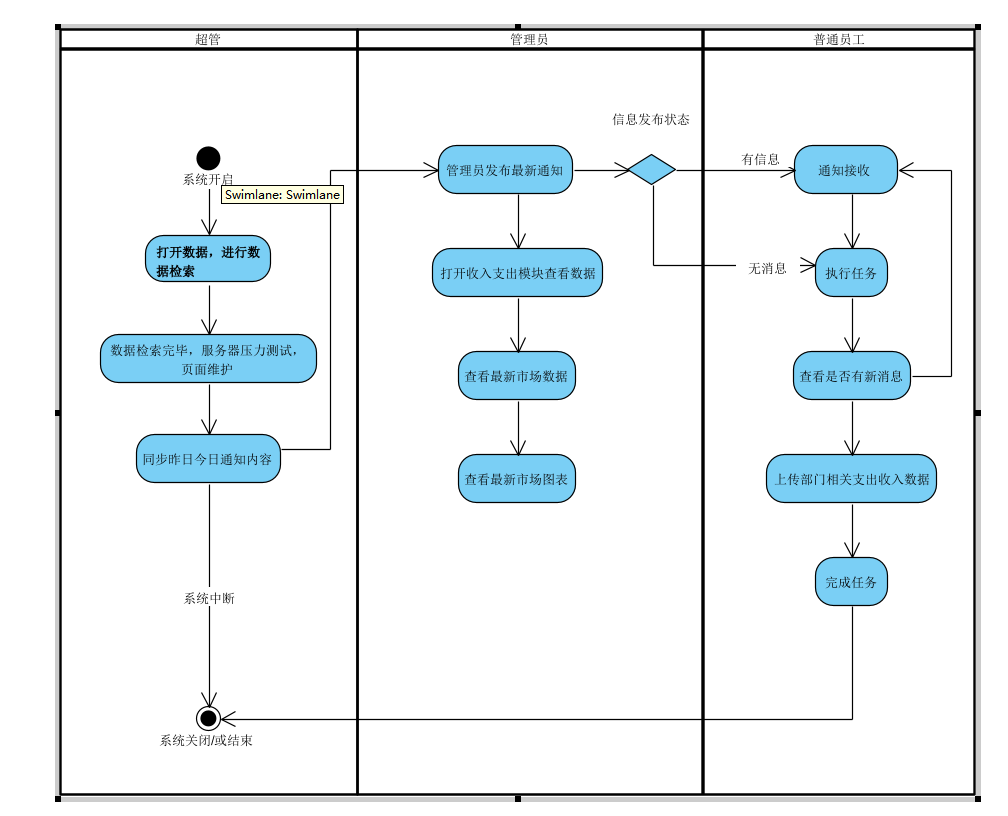


图4-10 不同角色操作活动图

普通用户在使用通知发布的时候，会被禁止使用，这也是唯一的功能上的区别，为了防止消息环境的破坏，我们增加了一层隔离。

c) 超级管理员SA：在简颢科技管理系统中，超级管理员的权限额外添加了数据库压力测试，系统维护等，其作用类似于DBA，对数据的操作较多，也是复杂度最大的一层，需要专业人士来对系统实现管控。那么SA的主要活动路径如下图4-11所示。

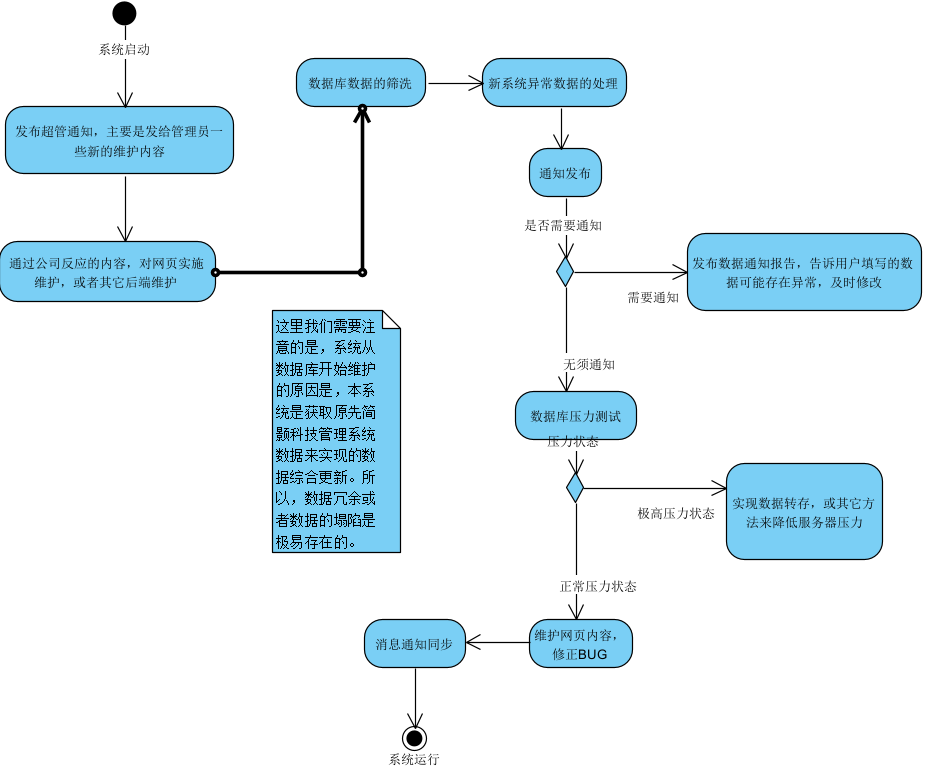


图4-11 SA主要操作活动图

1. 数据库设计

5.1 概念结构设计

系统中数据体系较大，且存在原公司系统的数据。这里使用属性图和ER图简单地展现一下新系统中包含的大部分重要内容。

图5-1所示的属性图为全部员工的属性数据信息，且只列举了重要的部分属性。

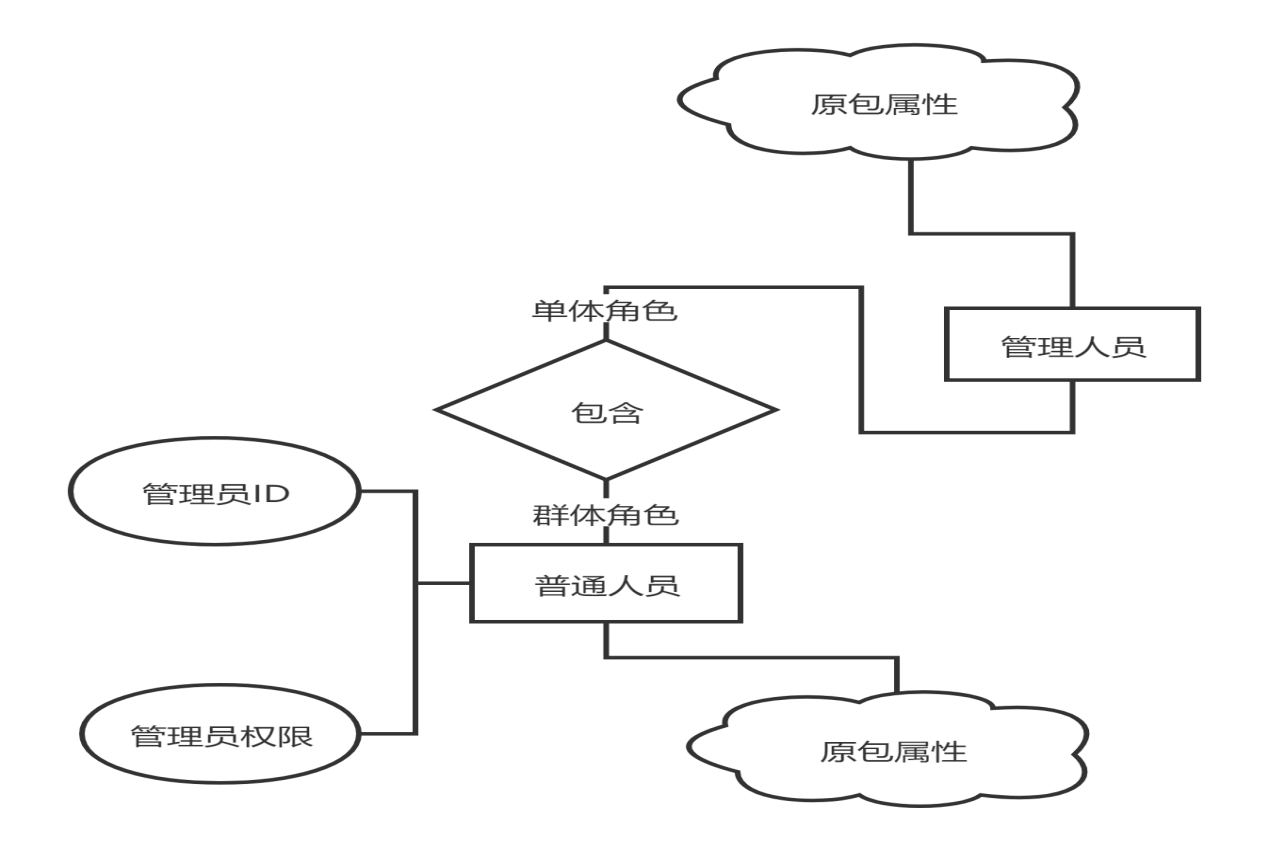
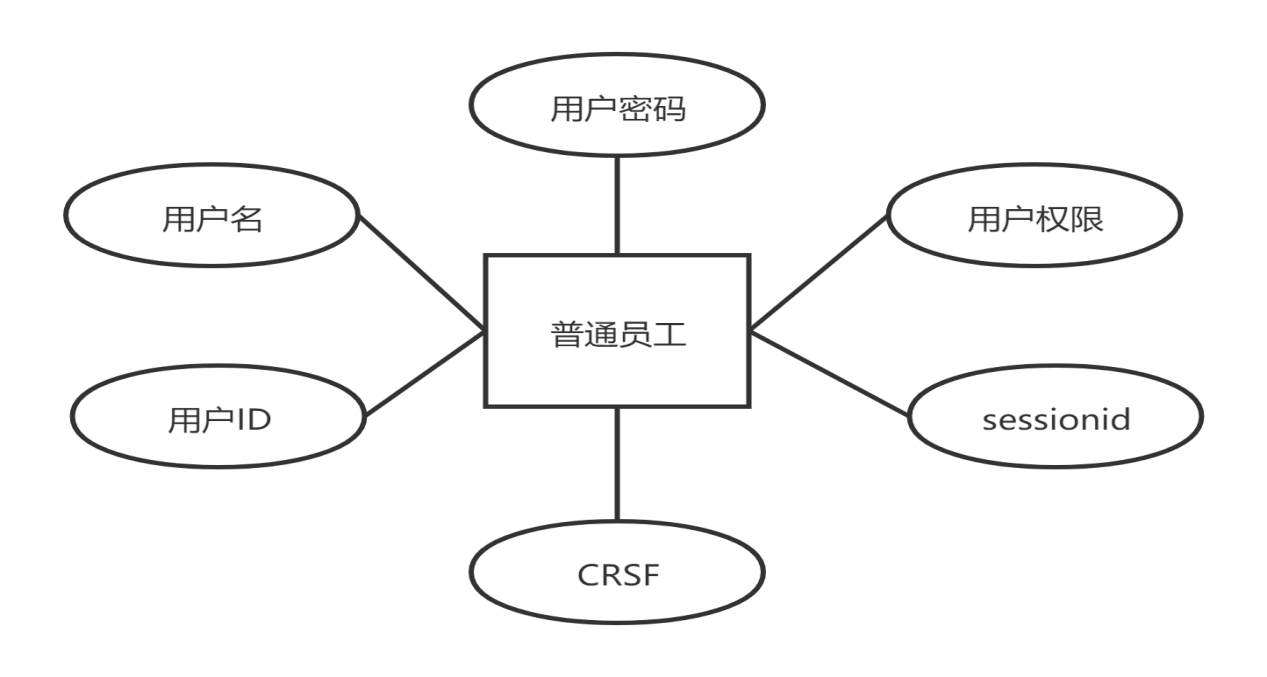


图5-1 简颢科技管理系统中普通员工及管理员的属性图

图5-2所示的ER图为通知发送所需要的信息收集与收入支出信息的关系，且只举例了部分重要的属性。

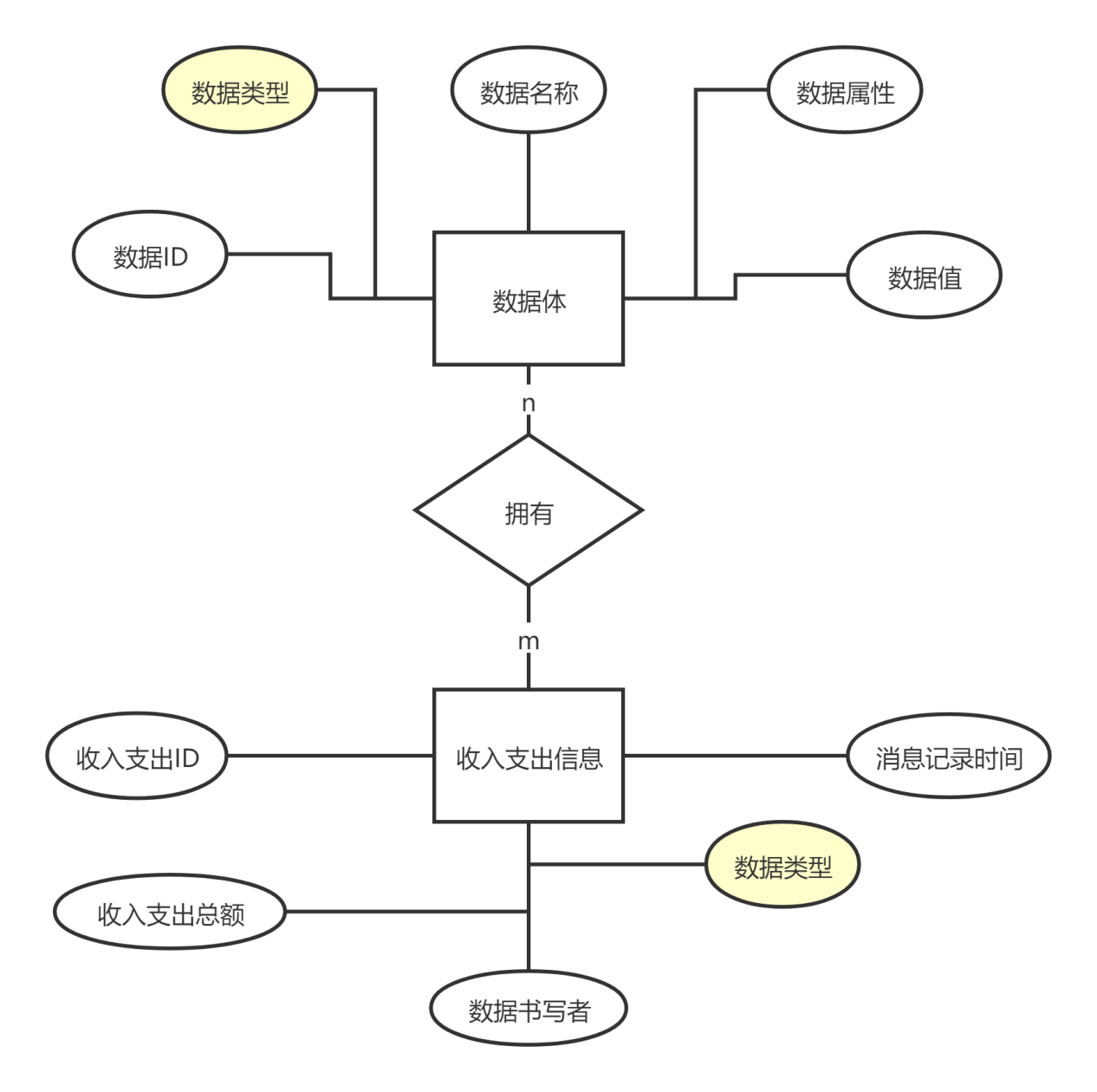


图5-2 简颢科技管理系统中信息收集与收入支出信息的ER图

5.2 逻辑结构设计

简颢科技管理系统的功能有很多，涉及了数据表项目众多，但系统对各表做了集成，使每个表都能具有相对分离相对聚合的效果。为了达到功能，并且能适配原先系统数据库的数据传递，我们使用了Python pymysql模块，来简化和利用mysql多表操作的方法，以达到表项最优化的状态。下面我会将数据库中常用的表项来做一个结构分析。

a) 用户数据表（用户ID，用户名，用户密码，用户权限，sessionID，csrfID）。用户数据表所对应的字段具体信息如表5-1所示。

b) 人员信息收集表（用户ID，用户姓名，用户民族，用户籍贯，用户出生地，用户性别，用户年月，用户健康状况，用户电话，用户邮箱，用户学历，用户经历，填写者，填写日期）。人员信息收集表所对应的字段具体信息如表5-2所示。

c) 公司收入信息表（项目ID，项目名称，项目总额，提交人员，提交时间，提交项目类别）。公司收入信息表所对应的字段具体信息如表5-3所示。

d) 公司支出信息表（项目ID，项目名称，项目总额，提交人员，提交时间，提交项目类别）。公司支出信息表所对应的字段具体信息如表5-4所示。

e) 产品项目表（产品ID，产品名称，产品单价，产品销售量，产品销售总额，产品销售日期）。产品项目表所对应的字段具体信息如表5-5所示。

f) 通知内容数据表（通知内容id，通知内容，通知提交用户名，通知更新时间）。通知内容数据表所对应的字段具体信息如表5-6所示。

表5-1 用户数据表（t\_user\_date）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 数据类型 | 类型长度 | 允许空 | 备注 |
| 用户ID | id | int | 0 | 否 | Primary |
| 用户名 | username | varchar | 20 | 否 | Unique |
| 用户密码 | password | varchar | 50 | 否 |  |
| 用户权限 | User\_PVLevel | int | 0 | 否 |  |
| sessionID | sessionid | varchar | 255 | 否 | Unique |
| csrfID | csrfid | varchar | 255 | 否 | Unique |

表5-2 员工信息表（t\_meminfolist）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 数据类型 | 类型长度 | 允许空 | 备注 |
| 用户ID | id | int | 0 | 否 | Primary |
| 用户姓名 | username | varchar | 20 | 否 |  |
| 用户民族 | nation | varchar | 20 | 否 |  |
| 用户籍贯 | NP | varchar | 20 | 否 |  |
| 用户出生地 | birthplace | varchar | 30 | 否 |  |
| 用户性别 | sex | varchar | 10 | 否 |  |
| 用户年月 | birthday | varchar | 30 | 否 |  |
| 用户健康状况 | health | varchar | 10 | 否 |  |
| 用户电话 | tel | varchar | 20 | 否 |  |
| 用户邮箱 | email | varchar | 50 | 否 |  |
| 用户学历 | education | varchar | 6 | 否 |  |
| 用户经历 | expText | varchar | 10000 | 否 |  |
| 填写者 | writername | varchar | 20 | 否 | Unique |
| 填写日期 | writedate | varchar | 100 | 否 |  |

表5-3 公司收入信息表（comecilist）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 数据类型 | 类型长度 | 允许空 | 备注 |
| 项目ID | id | int | 0 | 否 | Primary |
| 项目名称 | ProjectNM | varchar | 100 | 否 |  |
| 项目总额 | ProjectGot | Float | 0 | 否 |  |
| 提交人员 | writername | varchar | 20 | 否 |  |

续表5-3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 数据类型 | 类型长度 | 允许空 | 备注 |
| 提交时间 | writedate | varchar | 100 | 否 |  |
| 提交项目类别 | ProjectClass | varchar | 50 | 否 | 有默认 |

表5-4 公司支出信息表（comecolist）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 数据类型 | 类型长度 | 允许空 | 备注 |
| 项目ID | id | int | 0 | 否 | Primary |
| 项目名称 | ProjectNM | varchar | 100 | 否 |  |
| 项目总额 | ProjectCost | Float | 0 | 否 |  |
| 提交人员 | writername | varchar | 20 | 否 |  |
| 提交时间 | writedate | varchar | 100 | 否 |  |
| 提交项目类别 | ProjectClass | varchar | 50 | 否 | 有默认 |

表5-5 产品项目表（productionlist）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 数据类型 | 类型长度 | 允许空 | 备注 |
| 产品ID | id | int | 0 | 否 | Primary |
| 产品名称 | ProductionNM | varchar | 100 | 否 |  |
| 产品单价 | ProductionUnitPrice | Float | 0 | 否 |  |
| 产品售出数量 | ProductionSoldNum | Float | 0 | 否 |  |
| 产品总额 | ProductionPrice | Float | 0 | 否 | Auto |
| 产品售出时间 | ProductionSoldDate | varchar | 100 | 否 |  |

表5-6 通知内容数据表（infolist）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 数据类型 | 类型长度 | 允许空 | 备注 |
| 通知ID | id | int | 0 | 否 | Primary |
| 通知内容 | TextContent | varchar | 10000 | 否 |  |
| 发布者 | writername | varchar | 20 | 否 |  |
| 发布日期 | writedate | varchar | 100 | 否 |  |

1. 系统实现

6.1 登录入口功能的实现

a) 概述：简颢科技管理系统仅限于公司内部员工使用。账号的唯一性保密性是最为重要的，登录的表单提交安全性和账号密码及登录的检测机制，我都做了修改和集成，安全性更是在原先的post加密机制上，添加了新的安全算法，以保证数据的自保能力和可控能力。

b) 实现：系统通过对用户输入的cookie传值来判定其账号在近期是否存在某些登入状况，比如账号密码出错、csrfid失效、sessionid转换，或其他状态。其中csrfid用于防跨域攻击，以骗取原浏览器对用户登录后的信任。同时，在登录输入期间，我们修改了键盘输入的键位key值，反转为一种特定排序的新型键位key值以防止网页输入被记录键盘输入。

我们通过CSRF的响应文件部分（仅是CSRF中的一部分），引出我们的调用，由于Python高度集成了安全集，我们在页面中只需在有表单的位置前面加入｛CSRF\_token｝即可使用本框架，无需添加任何模块。

在实现了CSRF的防范机制后，引出我们的key值乱序实现，由于源码较长，我只展示key值修改的部分，其中我们需要用到的模块是PyHook。

登录入口界面效果如下图6-1所示：

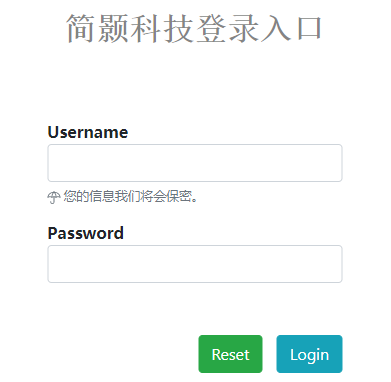


图6-1 登录界面图

6.2 人员数据收集功能实现

a) 概述：记录我司内部人员数据的相关信息。其包含新员工填写和老员工填写的检测模块。

b) 实现：人员数据收集表单，通过POST单表单发送至后端处理。

由于人员数据中表项的内容代码过长，不便展示，我们会在附录中具体展现。当我们填写玩人员数据表后，为了实现页面的局部更新，不刷新或清洗页面数据，我们使用AJAX来实现其操作。

实现后人员数据收集功能效果图如下图6-2所示：



图6-2人员数据收集效果图

6.3 公司财务部分功能实现

a) 概述：记录公司财务支出和收入的部分，实现了动态更新，实时更新，另外为了防止多人访问修改数据库导致的冲突或占线，我们对此引用了多线程的技术模块以及冲突封闭的技术模块。

b) 实现：公司财务部分包括的前端使用了AJAX的post方法。

由于本功能的特殊性，需要调用到原公司系统数据库的部分，此部分后端代码已隐藏，我们只展示数据获取至前端和时钟更新的技术源码。



另外我们展示一下防止数据库冲突和冲突封闭的技术。首先，Python多线程的模块支持如下， python主要是通过thread和threading这两个模块来实现对多线程支持，其效果也是不同的。python的thread模块是较为底层的模块，python的threading模块是对thread模块的一种封装，可以更方便的去使用。但是python（cpython）由于GIL的存在无法完美的使用threading来充分利用完全的CPU资源，如果想要发挥多核CPU的全部计算能力需要使用到python多线程模块中的线程核心支持，即multiprocessing模块，但是它对Win10的兼容性不太好，在本系统中，此代码应用在后端服务器中，做最终处理，服务器系统为Linux，所以兼容性问题可以不再考虑。

那么我是如何新建线程的呢？python3.x中摒弃了Python2.x中采用函数式thread模块中的start\_new\_thread()函数来产生原有线程的方式操作。

而在python3.x中我们通过threading模块创建了新的线程处理办法，主要有两种：一是通过threading.Thread(Target=executable Method)来传递给Thread对象一个可执行方法或者对象;第二种是继承threading.Thread定义子类并重写run()函数的办法。第二种办法中，唯一需要额外完成的就是必须重写方法run()。

至此，我们对线程处理的办法就很清晰了，下面我们看一下代码的实现部分。

这里我们展示的是第二种方法，即重写run的方法：

import threading

import time

class myThread(threading.Thread): # 继承父类threading.Thread

def \_\_init\_\_(self, threadID, name, counter):

threading.Thread.\_\_init\_\_(self)

self.threadID = threadID

self.name = name

self.counter = counter

def run(self): # 把要执行的代码写到run函数里面 线程在创建后会直接运行run函数

print("Starting " + self.name)

print\_time(self.name, self.counter, 5)

print("Exiting " + self.name)

def print\_time(threadName, delay, counter):

while counter:

time.sleep(delay)

print("%s process at: %s" % (threadName, time.ctime(time.time())))

counter -= 1

# 创建新线程

thread1 = myThread(1, "Thread-1", 1)

thread2 = myThread(2, "Thread-2", 2)

# 开启线程

thread1.start()

thread2.start()

# 等待线程结束

thread1.join()

thread2.join()

到这里，我们就解决了多用户访问时出现的冲突，那么数据库访问有时是需要同步执行的比如在公司财务模块中我们如果需要多个用户对某个数据修改，那么为了保证正确性，我们需要对多线程进行同步操作。下面是我对这种问题的解决办法。

使用Thread对象的Lock和Rlock可以实现一些较为简单的线程同步，这两个对象都具有acquire和release这两种方法，对于某些一次只能对一个数据进行操作的线程流，可以将这一步操作放至到acquire和release方法中间，已解决这一问题。

但需要注意的是，Python有一个GIL机制即Global Interpreter Lock，在任意线程执行前我们都必须获取它才能执行，每执行完100条字节码数据后，全局锁才会释放，切换到其他线程进行执行。多线程实现同步有四种方式：锁机制，信号量，条件判断和同步队列。下面我主要使用到的是两种同步机制：锁机制和同步队列。

源码实现如下：

import threading

import time

class myThread(threading.Thread):

def \_init\_(self.threadID, name, counter):

threading.Thread.\_init\_(self)

self.threadID = threadID

self.name = name

self.counter = counter

def run(self):

print("Starting"+ self.name)

# 获得锁，成功获得锁定后返回 True

# 可选的 timeout 参数不填时将一直阻塞直到获得锁定

# 否则超时后将返回 False

threadLock.acquire()

print\_time(self.name, self.counter, 5)

# 释放锁

threadLock.realease()

def print\_time(threadName, delay,counter):

while counter:

time.sleep(delay)

print"%s: %s"%(threadName,time.ctime(time.time())))

counter -= 1

至此，我们数据库冲突的问题也实现了。在本功能中，我们会使用到上面所展示的方法，这里我们不在展示，下面我们展示更新前后的对比图，如图6-3所示：



图6-3-1公司财务部分更新前效果图



图6-3-2 公司财务部分更新后效果图

6.4 市场模块功能实现

a) 概述：市场功能主要实现了对本年度中最高销售额的十种产品进行展示，并且具有产品录入的功能，详细记录了项目的数据，并实现同步更新的功能。

b) 实现：市场模块前端使用AJAX方法get实现。

市场盈亏模块效果图如下图6-4所示：



图6-4 市场模块效果图

6.5 数据分析及可视化模块实现

a) 概述：本模块是本系统中最核心的部分，主要是对收集到的数据进行整合清洗，并通过动态图表展示的方式以呈现前十销售额商品的信息。这个模块后端我们引入了Python Math、Pandas、matplotlib等数据分析模块，和数据预测模块、线性回归算法及seaborn模块。前端数据图表显示使用了ercharts和ecStat两个JS文件。

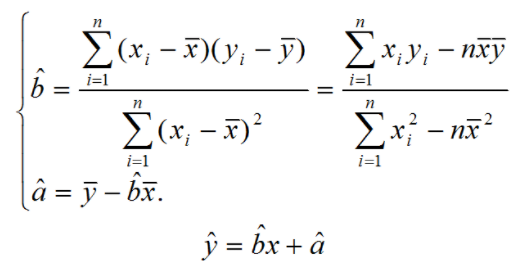
b) 实现：数据预测及可视化模块预测图表按钮触发AJAX。

下面就是我们本功能的核心部分数据分析的后端实现部分。我们会收集到大量的公司数据和本系统所新增的数据，下面就按照数据获取-数据存储与提取-数据预处理-数据建模与分析-数据可视化的步骤来展示整个状态。

首先是数据的预处理，我们使用了pandas模块，对本系统数据库中的数据进行筛选，选出我们需要的各产品市场销售额及产品类别，及通过数据超市，我这里使用的百度API数据，获取到的当前市场部分数据产品的销售额等信息。这样一个简单的比例关系就有了，使用pandas和pymysql获取数据。

那么，在详细说到比例关系之前，我们先引入线性OLS回归的概念，本系统通过此方法根据历史对数收益率预测市场运动方向，注意这里只是知道方向，根据数学知识，我们知道线性回归需要的是X和Y的散点图，然后通过最小二乘法来找出最贴切的直线关系。

公式如下所示：



这里我们知道了数据只需要按照这样的公式来执行就能拟定出一条函数线来，但是XY的值是什么我们怎么获取呢？本系统定义了，使用系统数据本产品销售额作为Y轴数据，而X采用的市场占比随机因子。

下面我们来说一下市场占比随机因子的获取方法，比如，本地的一个产品‘Ai视觉技术支持’，获取到整个市场中同类产品的数据及同月销售额，我们做一个比例。自己的销售额与同类产品的数据销售额做比，同类做分子，自己的销售额做分母（一般比例在万级以上），下面引用到这个比例都会使用XY比来表示。

至此，数据的X，Y都已经有了数值，但是一个产品不代表整个公司的所有产品走势，所以这里我们引入了X通量，即所有产品的类型得到单个XY比，合并加权平均后，就可以得到一个商品在比例中的权值，这样的带入原先的XY比后，就能具有全产品的代表性。再将每一个产品的散点放入计算，就能得到未来公司收入的推演方程线了。回归模型建立，后面会使用newdata来直接获取plt数据，来将它出入到前端，实现erchart绘制，而不使用matplotlib。

这样，整个模块功能实现就完成了，下面会把后端处理的数据转会AJAX，并通过erchart实现绘图。

数据分析及可视化模块效果如图6-6所示：

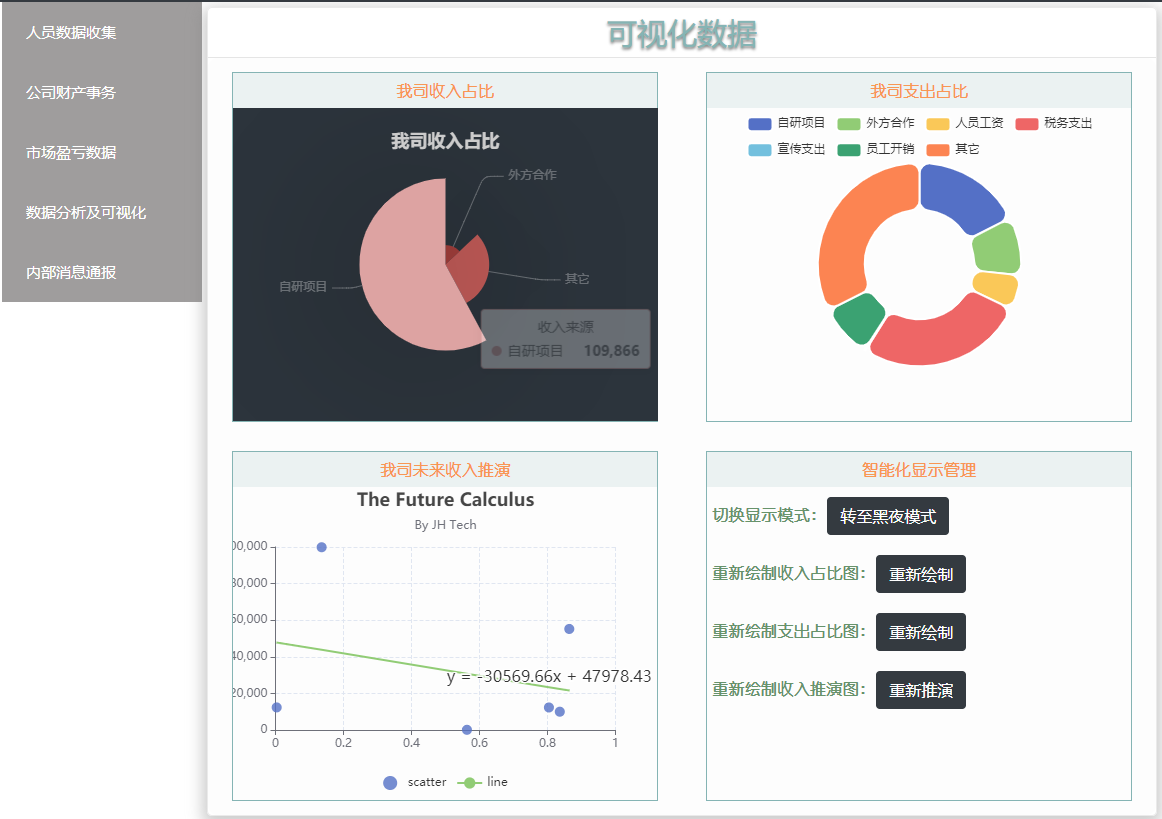


图6-6 AJAX数据返回图

6.6 消息通知功能实现

a) 概述：内部消息管理，需要管理员每日发送通知或命令来告知普通员工，普通员工可以查看最近两天的消息。

b) 实现：消息通知功能前端AJAX+DDBI。

具体效果如下图6-7所示。



图6-7 消息通知模块效果图

1. 系统调试

7.1 首页登录界面的调试

测试用例如表7-1所示。

表7-1 登陆测试用例表

|  |  |  |
| --- | --- | --- |
| 功能描述 | 系统用户登陆 | |
| 用例目的 | 用户登陆是否有错误提示 | |
| 必要条件 | 点击“Login”按钮 | |
| 输 入 | 期望值 | 实际情况 |
| 不输入用户名 | 提示“用户名或密码为空” | 提示“用户名或密码为空” |
| 不输入密码 | 提示“用户名或密码为空” | 提示“用户名或密码为空” |
| 输入错误用户名或密码 | 请输入正确的用户名和密码 | 请输入正确的用户名和密码 |
| 输入正确用户名和密码 | 无提示，登陆成功页面跳转 | 无提示，登陆成功页面跳转 |

表7-2 首页显示测试用例表

|  |  |
| --- | --- |
| 功能A描述 | 系统用户首页 |
| 用例目的 | 用户不登陆能否查看主界面以及登陆后是否可以正常查看 |
| 必要条件 | 页面跳转 |

表7-2 首页显示测试用例表续表

|  |  |  |
| --- | --- | --- |
| 操 作 | 期望值 | 实际情况 |
| 不登陆，直接网址跳转 | 跳回登陆页面并提示问题 | 跳回登陆页面并提示问题 |
| 登陆成功自动跳转 | 跳转到主页，可使用功能 | 跳转到首页，可使用功能 |

电脑端登陆成功后默认进入首页，首页可以查看用户登录标识及使用相关功能如图7-1所示。

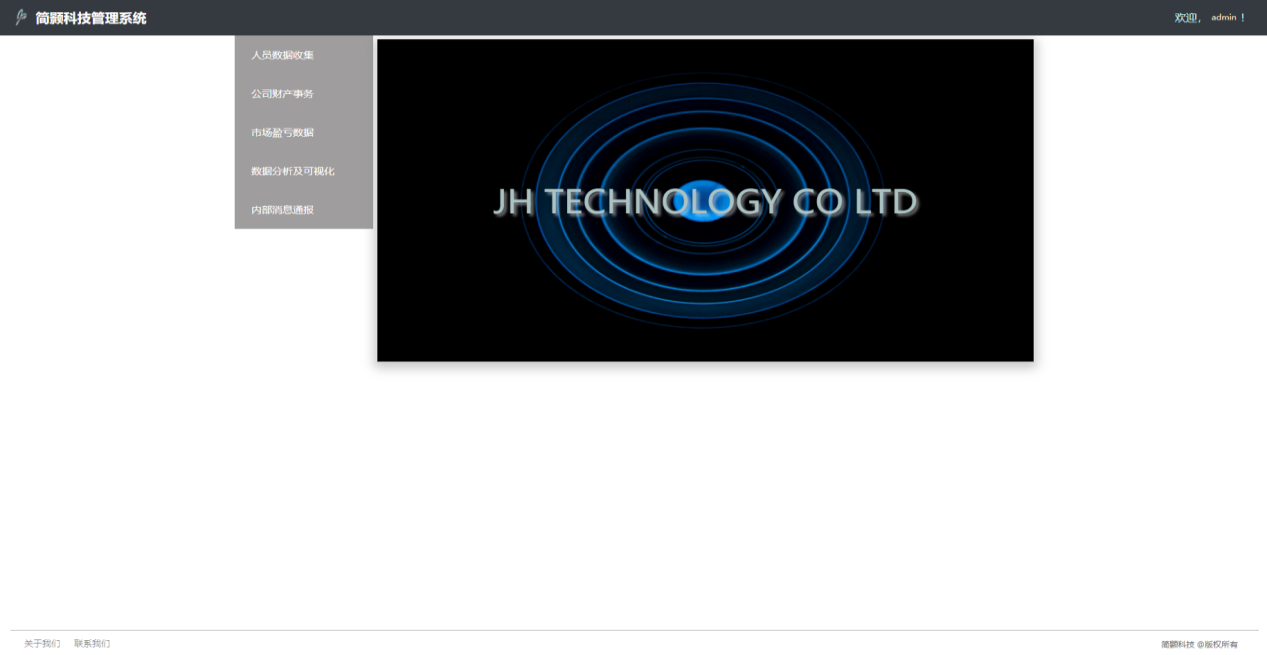


图7-1-1 主页展示图



图7-1-2 主页及功能调用图

右上角可以查看系统的登录者用户名。如下图7-2所示。

用户名显示图

图7-2 用户名显示图

7.2 人员数据收集的调试

测试用例如表7-3所示。

表7-3 人员数据收集用例表

|  |  |  |
| --- | --- | --- |
| 功能描述 | 人员数据的云端校验 | |
| 用例目的 | 人员数据新建或更新测试 | |
| 必要条件 | 点击“submit”按钮 | |
| 输 入 | 期望值 | 实际情况 |
| 新用户输入 | 提示“您的信息已创建！” | 提示“您的信息已创建！” |
| 老用户更新 | 提示“您的信息已更新！” | 提示“您的信息已更新！” |
| 更新或新建出错 | 提示“出错，请询问管理员” | 提示“出错，请询问管理员” |

创建或更新人员数据需要填写数据。具体如图7-3所示。



图7-3 人员数据收集填写图

更新成功及数据库更新情况如下图7-4所示。



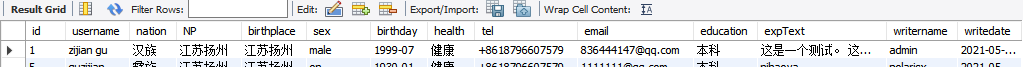


图7-4 人员数据更新及数据库表状态图

7.3 公司财务功能调试

测试用例如表7-4所示。

表7-4 公司财务测试用例表

|  |  |  |
| --- | --- | --- |
| 功能描述 | 各分支收入支出动态显示 | |
| 用例目的 | 校验该功能是否存在显示值出错或不显示的问题 | |
| 必要条件 | 各部门普通员工或管理员可执行 | |
| 查 看 | 期望值 | 实际值 |
| 收入或支出项目提交 | 提示“这是一个新项目” | 提示“这是一个新项目” |
| 收入或支出项目更新 | 提示“已更新项目” | 提示“已更新项目” |

当用户在点击提交或是别的用户更新完数据后，各个分支都会做出相应的变动，已达到动态刷新。如图7-5所示。



图7-5 收支动态刷新图

当用户提交支出或收入项后提示。如图7-6所示。

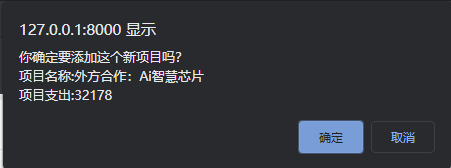


图7-6-1 收支确认图

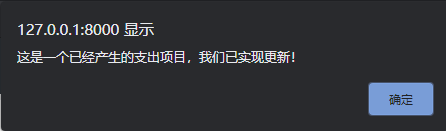


图7-6-2 收支更新图

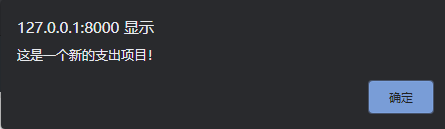


图7-6-3 收支新建图

7.4 市场功能的调试

测试用例如表7-5所示。

表7-5 市场功能测试用例表

|  |  |  |
| --- | --- | --- |
| 功能描述 | 产品项目更新或建立 | |
| 用例目的 | 查看是否分离新项目和原有项目 | |
| 操 作 | 期望值 | 实际情况 |
| 更新产品项目 | 提示“产品信息已更新” | 提示“产品信息已更新” |
| 新建产品项目 | 提示“产品项目已建立” | 提示“产品项目已建立” |

当用户在点击提交或是别的用户更新完数据后，各个分支都会做出相应的变动，已达到动态刷新。如图7-7所示。



图7-7 市场盈亏数据功能图

产品新建或更新后提示如下图7-8所示。

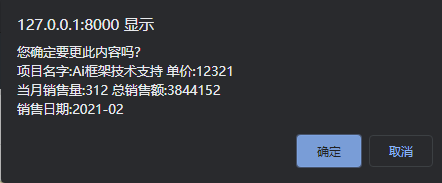


图7-8-1 提交项目信息提示图

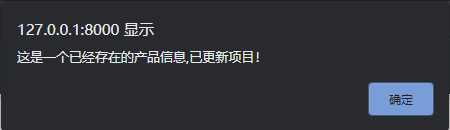


图7-8-2 更新项目提示图

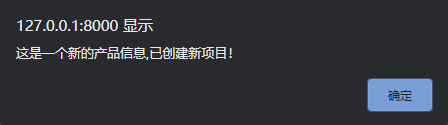


图7-8-3 新建项目提示图

7.5 数据分析及可视化功能的调试

测试用例如表7-6所示。

表7-6 数据分析及可视化测试用例表

|  |  |  |
| --- | --- | --- |
| 功能描述 | 图表重绘 | |
| 用例目的 | 测试图表绘制按钮有效性 | |
| 必要条件 | 系统中数据体量达到15条以上 | |
| 操 作 | 期望值 | 实际情况 |
| 点击重新绘制按钮 | 相应图表重新绘制 | 相应图表重新绘制 |

当我们点击重新绘制按钮时，会实现图表的重新模拟，并显示在相应区域内。效果图如下，本处仅展示数据推演部分图表重绘。如图7-9所示。

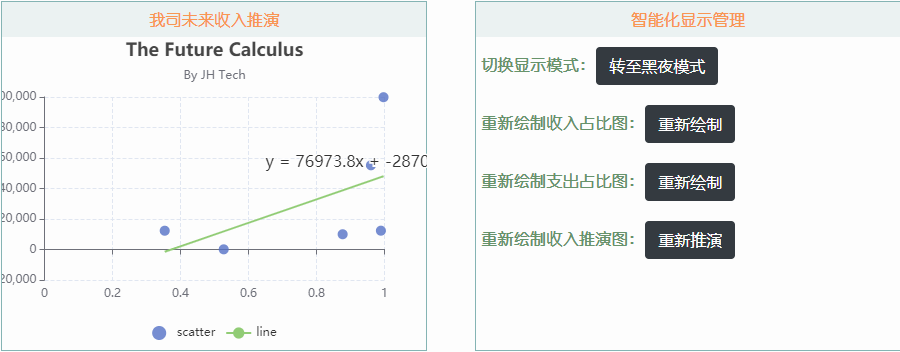


图7-9 新建项目提示图

7.6 通知功能的调试

测试用例如表7-7所示。

表7-7 通知功能测试用例表

|  |  |  |
| --- | --- | --- |
| 功能描述 | 通知发送 | |
| 用例目的 | 测试是否可以越权发送或发送是否有误 | |
| 必要条件 | 权限设置 | |
| 操 作 | 期望值 | 实际情况 |
| 点击发送按钮 | 提示“消息同步成功” | 提示“消息同步成功” |
| 无权发送 | 提示“您是普通员工不可发送”，并且锁住填写和发送按钮 | 提示“您是普通员工不可发送”，并且锁住填写和发送按钮 |

当我们点击发送按钮时，会检索当前用户是否具有发送权限，如果有，会同步消息内容，而如果没有，会自动封闭其发送消息框和发送按钮的使用权限。具体如图7-10所示。



图7-10 消息发送显示图

当我们使用普通用户来发送消息时，会提示越权发送，而被禁止使用通知框和发布按钮。具体见图7-11所示。



图7-11 消息越权发送失败图

1. 结束语

至此，简颢科技管理系统已基本完成。在整个项目中我遇到各种各样的问题，在最后结束语中总结一下，希望能够对自己有所收益。首先，就是在部署整个Django框架时分离调试阶段和发布阶段，在这个阶段中，我遇到了CSS和JS文件无法导入的问题，也困扰了我很久。通过stackoverflow上的交流得知需要在Django中的setting文件中设置DEBUG属性为False以及SECURE\_CONTENT\_TYPE\_NOSNIEF的属性设置为False属性，而且要添加url(r'^static/(?P<path>.\*)$', static.serve,{'document\_root': settings.STATIC\_ROOT}, name='static')在url路径中。这样项目就能在发布状态访问JS和CSS内容了。

另外遇到使用JS删除自身的问题，以及设置标准时间的方法，还有使用JQuery来获取radio单选框的值问题，这些都是一些细节问题。

在css中，我们如果遇到div中的内容很长如div{1000个‘字符’}，我们应该怎么处理呢？首先我们会想到的是overflow:scroll的方法，但是，我们不想要横向的滚动条，怎么办呢？那么，我们可以使用word-break: break-all来实现自动转行，而不会出现横向滚动条了。

以上问题是我本次系统制作中遇到的主要问题，当然也有些许没有完善的功能，比如图表查看的权限划分，以及人员数据收集的定时提醒功能，这样系统可以更加集成，也是对管理员工作的一种减负。

**参考文献**

[1](美)斯图尔特·里杰斯（Stuart Reges）,马蒂·斯特普（Marty Stepp）,艾利森·奥伯恩（Allison Obourn）.Python程序设计与算法思维[M].北京:机械工业出版社,2020.

[2]王青天,孔越.Python金融大数据风控建模实战：基于机器学习[M].北京:机械工业出版社,2020.

[3]勇闯天涯的虾客.Python多线程[DB].新浪博客,2018.

[4]丹尼尔·陈（Daniel Y. Chen）.Python数据分析 活用Pandas库[M].北京:人民邮电出版社,2020.

[5]吴仲治.python开发技术大全[M].北京:机械工业出版社,2020.

[6]宋天龙.python数据分析与数据化运营[M].北京:机械工业出版社,2018.

[7]朱洁.大数据架构详解：从数据获取到深度学习[M].北京:电子工业出版社,2016.

[8]胡阳.django企业开发实战 高效python web框架指南[M].北京:人民邮电出版社,2019.

[9] Tian H , Zhao J , Shen J . Research on Optimized Storage and Analysis System of Web Log Based on Django's MVC Framework[J]. Journal of Physics: Conference Series, 2021, 1769(1):012065 (9pp).

[10]王伊, 王韶红, 刘晋泽,等. Vue.js与Django组合框架的网络社交系统单页面架构方案设计[J]. 信息技术与信息化, 2020, 238(01):127-129.

**致 谢**

在零碎的三个月开发期间，我学到了很多，通过查阅资料、询问老师、询问同学，社区交流，我顺利完成了系统的设计与开发。在这里我要感谢所有为我提供过帮助的人。

首先是我的老师同学以及我的学校。学校这四年给我们设定了良好的学习线路，开设了很多课程，为我在计算机的学习道路上做了很多铺垫。在本系统的设计和开发中我也遇到了很多问题，包含技术问题，设计问题，学术问题，但不仅限于此，李运平老师也给予了用心指导。老师在这三个月的开发期间也很关照我，指导我们开展毕业项目的注意事项，包括毕设的格式，书写内容所需要注意的点点滴滴，这很大程度上帮助了我完成了毕业设计。

同学在我完成系统设计的期间，指导过我很多，比如Python模块的使用，告诉我JS的云链接，以及Django框架的讲解，以帮助我对整个web框架的理解。

其次是公司的技术支持。公司为我的毕设提供了数据库及服务器支持，让我能远程使用大量的数据。在此之上，公司提供了网站发布支持，通过使用内网穿透和Nginx (engine x)反向代理web服务器技术支持，同时也提供了IMAP/POP3/SMTP服务。

最后，需要感谢的是我的父母。他们对我的默默支持和鼓励，由衷的感谢。另外，需要感谢的是表哥对本系统提供的安全技术支持，帮我完成了XXS、SQL注入、CSRF跨域攻击等安全技术的帮助。

**附 录**

附录1 源程序清单

附录1.1 项目前端主要模块文件（依赖注入）

<script src="https://cdn.bootcdn.net/ajax/libs/jquery/3.6.0/jquery.min.js"></script>  
<script src="https://cdn.staticfile.org/jquery-cookie/1.4.1/jquery.cookie.min.js"></script>  
<script src="/static/JS/jquerysession.js"></script>  
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>  
<script src="https://cdn.bootcdn.net/ajax/libs/twitter-bootstrap/4.6.0/js/bootstrap.bundle.min.js"></script>  
<link href="https://cdn.bootcdn.net/ajax/libs/twitter-bootstrap/4.6.0/css/bootstrap.min.css" rel="stylesheet">  
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/

bootstrap-icons@1.4.1/font/bootstrap-icons.css">  
<link rel="stylesheet" href="/static/css/mycsslist\_ShowYourInfo.css">  
<script src="/static/JS/echarts.min.js"></script>  
<script src="/static/JS/ecStat.min.js"></script>  
<script src="/static/JS/MyJsList.js"></script>

附录1.2 项目后端处理文件（所有处理内容，不包含跨系统数据库内容）

import datetime  
import json  
import random  
  
from django.forms import model\_to\_dict  
from django.http import HttpResponse, JsonResponse  
from django.shortcuts import render, redirect  
from django.views.decorators.csrf import csrf\_exempt  
  
from DJ\_App.models import user\_date, memInfoList, COMECOList, COMECIList, ProductionList, InfoList, \  
 administratorNameList  
  
  
# 主页处理  
def index(request):  
 res = render(request, 'MainPage.html')  
 UPTip = request.COOKIES.get('UPerror')  
 if UPTip == "":  
 res.set\_cookie('UPerror', -1)  
 return res  
 else:  
 return res  
  
  
# 系统页面处理  
def showInfo(request):  
 resBack = render(request, 'MainPage.html')  
 UPTip = request.COOKIES.get('UPerror')  
 UserName = request.session.get('userName')  
 content = {}  
 content['username'] = UserName  
 if UPTip == "":  
 resBack.set\_cookie('UPerror', -1)  
 return resBack  
 elif UPTip == '0':  
 res = render(request, 'ShowYourInfo.html', content)  
 return res  
 else:  
 resBack.set\_cookie('UPerror', -1)  
 return resBack  
  
  
# 主页登录表单处理  
def testNameandPass(request):  
 name = request.POST.get("username")  
 passwd = request.POST.get("password")  
 content = {'username': name, 'password': passwd}  
  
 if request.COOKIES.get('UPerror') == '0':  
 resJump = render(request, 'ShowYourInfo.html', content)  
 return resJump  
  
 if name == "" or passwd == "":  
 res = redirect('../')  
 res.set\_cookie('UPerror', -1)  
 return res  
 try:  
 if user\_date.objects.get(username=name):  
 print("right username!")  
 except user\_date.DoesNotExist:  
 print("name is error!")  
 res = redirect('../')  
 res.set\_cookie('UPerror', 1)  
 return res  
 else:  
 try:  
 if user\_date.objects.get(username=name, password=passwd):  
 print("right name and pass!")  
 except user\_date.DoesNotExist:  
 print("error pass!")  
 res = redirect('../')  
 res.set\_cookie('UPerror', 2)  
 return res  
 else:  
 res = redirect('../info/')  
 res.set\_cookie('UPerror', 0)  
 request.session['userName'] = content['username']  
 return res  
  
  
# 人员信息表单处理  
@csrf\_exempt  
def textMemInfo(request):  
 newdata = {  
 'username': request.POST.get('username'),  
 'nation': request.POST.get('nation'),  
 'NP': request.POST.get('NP'),  
 'birthplace': request.POST.get('birthplace'),  
 'sex': request.POST.get('sex'),  
 'birthday': request.POST.get('birthday'),  
 'health': request.POST.get('health'),  
 'tel': request.POST.get('tel'),  
 'email': request.POST.get('email'),  
 'education': request.POST.get('education'),  
 'expText': request.POST.get('expText'),  
 'writername': request.session.get('userName'),  
 'writedate': str(datetime.datetime.now())  
 }  
 print(newdata['sex'])  
 try:  
 getDateObj = memInfoList.objects.get(writername=newdata['writername'])  
 # getDateObjList= memInfoList.objects.filter(nation='蒙古族')  
 except memInfoList.DoesNotExist:  
 print('没有查询到相关数据！')  
 newdata['Message'] = "您的信息已创建！"  
 newDatalist = memInfoList(username=newdata['username'], nation=newdata['nation'],  
 NP=newdata['NP'],  
 birthplace=newdata['birthplace'], sex=newdata['sex'],  
 birthday=newdata['birthday'],  
 health=newdata['health'], tel=newdata['tel'],  
 email=newdata['email'],  
 education=newdata['education'], expText=newdata['expText'],  
 writername=newdata['writername'], writedate=newdata['writedate'])  
 newDatalist.save()  
 #下面是对数据库更新的else部分  
 else:  
 print('查询到相关数据！')  
 newdata['Message'] = "您的信息已更新！"  
 getDateObj.username = newdata['username']  
 getDateObj.nation = newdata['nation']  
 getDateObj.NP = newdata['NP']  
 getDateObj.birthplace = newdata['birthplace']  
 getDateObj.sex = newdata['sex']  
 getDateObj.birthday = newdata['birthday']  
 getDateObj.health = newdata['health']  
 getDateObj.tel = newdata['tel']  
 getDateObj.email = newdata['email']  
 getDateObj.education = newdata['education']  
 getDateObj.expText = newdata['expText']  
 getDateObj.writername = newdata['writername']  
 getDateObj.writedate = newdata['writedate']  
 getDateObj.save()  
 return HttpResponse(json.dumps(newdata))  
 # 这里我们要做一个数据库的检测，是否相同用户名填写的数据，我们要做一个及时更新。  
  
  
# 获取自研项目、外方合作、人员工资、税务支出、宣传支出、员工开销等字样的索引值  
def getSpecialString(string):  
 newStrArr = ['自研项目', '外方合作', '人员工资', '税务支出', '宣传支出', '员工开销']  
 getIndex = -1  
 for k in range(0, len(newStrArr)):  
 if string.find(newStrArr[k]) != -1:  
 getIndex = k  
 break  
  
 if getIndex == 0:  
 getIndexStr = '自研项目'  
 elif getIndex == 1:  
 getIndexStr = '外方合作'  
 elif getIndex == 2:  
 getIndexStr = '人员工资'  
 elif getIndex == 3:  
 getIndexStr = '税务支出'  
 elif getIndex == 4:  
 getIndexStr = '宣传支出'  
 elif getIndex == 5:  
 getIndexStr = '员工开销'  
 else:  
 getIndexStr = '其它'  
 return getIndexStr  
  
  
# 获取自研项目、外方合作等字样的索引值  
def getSpecialString2(string):  
 newStrArr = ['自研项目', '外方合作']  
 getIndex = -1  
 for k in range(0, len(newStrArr)):  
 if string.find(newStrArr[k]) != -1:  
 getIndex = k  
 break  
  
 if getIndex == 0:  
 getIndexStr = '自研项目'  
 elif getIndex == 1:  
 getIndexStr = '外方合作'  
 else:  
 getIndexStr = '其它'  
 return getIndexStr  
  
  
# 公司财务支出表单发布/更新  
@csrf\_exempt  
def ECOOFormUpdate(request):  
 newdata = {  
 'projectNM': request.POST.get('projectNM'),  
 'projectCost': request.POST.get('projectCost'),  
 'writername': request.session.get('userName'),  
 'writedate': str(datetime.datetime.now()),  
 }  
 IsExistClass = getSpecialString(newdata['projectNM'])  
 if IsExistClass == '其它':  
 print('这是一个其它类型，未找到索引！')  
 newdata['projectClass'] = IsExistClass  
  
 else:  
 print('这是一个''+IsExistClass+''！')  
 newdata['projectClass'] = IsExistClass  
  
 try:  
 getDateObj = COMECOList.objects.get(ProjectNM=newdata['projectNM'])  
 except COMECOList.DoesNotExist:  
 print('这是一个新的支出项目')  
 newdata['Message'] = '这是一个新的支出项目！'  
 newDatalist = COMECOList(ProjectNM=newdata['projectNM'], ProjectCost=float(newdata['projectCost']),  
 writername=newdata['writername'], writedate=newdata['writedate'],  
 ProjectClass=newdata['projectClass'])  
 newDatalist.save()  
 return HttpResponse(json.dumps(newdata))  
 #修改数据库部分，其中包含原数据库信息不便展示  
 else:  
 print('这是一个已经产生的支出项目')  
 newdata['Message'] = '这是一个已经产生的支出项目，我们已实现更新！'  
 getDateObj.ProjectCost = float(newdata['projectCost'])  
 getDateObj.ProjectClass = newdata['projectClass']  
 getDateObj.writername = newdata['writername']  
 getDateObj.writedate = newdata['writedate']  
 getDateObj.save()  
 return HttpResponse(json.dumps(newdata))  
  
  
# 公司财务收入表单发布/更新  
@csrf\_exempt  
def ECOIFormUpdate(request):  
 newdata = {  
 'ProjectNM': request.POST.get('ProjectNM'),  
 'ProjectGot': request.POST.get('ProjectGot'),  
 'writername': request.session.get('userName'),  
 'writedate': str(datetime.datetime.now()),  
 }  
 IsExistClass = getSpecialString2(newdata['ProjectNM'])  
 if IsExistClass == '其它':  
 print('这是一个其它类型，未找到索引！')  
 newdata['projectClass'] = IsExistClass  
  
 else:  
 print('这是一个''+IsExistClass+''！')  
 newdata['projectClass'] = IsExistClass  
  
 try:  
 getDateObj = COMECIList.objects.get(ProjectNM=newdata['ProjectNM'])  
 except COMECIList.DoesNotExist:  
 print('这是一个新的收入项目')  
 newdata['Message'] = '这是一个新的收入项目！'  
 newDatalist = COMECIList(ProjectNM=newdata['ProjectNM'], ProjectGot=float(newdata['ProjectGot']),  
 writername=newdata['writername'], writedate=newdata['writedate'],  
 ProjectClass=newdata['projectClass'])  
 newDatalist.save()  
 return HttpResponse(json.dumps(newdata))  
 else:  
 print('这是一个已经产生的收入项目')  
 newdata['Message'] = '这是一个已经产生的收入项目，我们已实现更新！'  
 getDateObj.ProjectGot = float(newdata['ProjectGot'])  
 getDateObj.ProjectClass = newdata['projectClass']  
 getDateObj.writername = newdata['writername']  
 getDateObj.writedate = newdata['writedate']  
 getDateObj.save()  
 return HttpResponse(json.dumps(newdata))  
  
  
# 公司财务的动态更新  
@csrf\_exempt  
def ECOIOListUpdate(request):  
 newdata = {}  
 # 这里我们获取支出的相关项目  
 # 获取自研项目0、外方合作1、人员工资2、税务支出3、宣传支出4、员工开销5、其它-1  
 getECOO\_0 = COMECOList.objects.filter(ProjectClass='自研项目')  
 getECOO\_1 = COMECOList.objects.filter(ProjectClass='外方合作')  
 getECOO\_2 = COMECOList.objects.filter(ProjectClass='人员工资')  
 getECOO\_3 = COMECOList.objects.filter(ProjectClass='税务支出')  
 getECOO\_4 = COMECOList.objects.filter(ProjectClass='宣传支出')  
 getECOO\_5 = COMECOList.objects.filter(ProjectClass='员工开销')  
 getECOO\_sub1 = COMECOList.objects.filter(ProjectClass='其它')  
 ###########################################################  
 # 这里我们获取收入的相关项目  
 # 获取自研项目0、外方合作1、其它-1  
 getECOI\_0 = COMECIList.objects.filter(ProjectClass='自研项目')  
 getECOI\_1 = COMECIList.objects.filter(ProjectClass='外方合作')  
 getECOI\_sub1 = COMECIList.objects.filter(ProjectClass='其它')  
  
 ############################################################  
 # 这里是支出计量  
 # 获取到6类项目的支出情况（自研项目0、外方合作1、人员工资2、税务支出3、宣传支出4、员工开销5、其它-1）  
  
 ECOOsum0 = 0  
 for x in range(0, len(getECOO\_0)):  
 ECOOsum0 += getECOO\_0[x].ProjectCost  
  
 ECOOsum1 = 0  
 for x1 in range(0, len(getECOO\_1)):  
 ECOOsum1 += getECOO\_1[x1].ProjectCost  
  
 ECOOsum2 = 0  
 for x2 in range(0, len(getECOO\_2)):  
 ECOOsum2 += getECOO\_2[x2].ProjectCost  
  
 ECOOsum3 = 0  
 for x3 in range(0, len(getECOO\_3)):  
 ECOOsum3 += getECOO\_3[x3].ProjectCost  
  
 ECOOsum4 = 0  
 for x4 in range(0, len(getECOO\_4)):  
 ECOOsum4 += getECOO\_4[x4].ProjectCost  
  
 ECOOsum5 = 0  
 for x5 in range(0, len(getECOO\_5)):  
 ECOOsum5 += getECOO\_5[x5].ProjectCost  
  
 ECOOsumsub1 = 0  
 for x6 in range(0, len(getECOO\_sub1)):  
 ECOOsumsub1 += getECOO\_sub1[x6].ProjectCost  
  
 ECOOsum = ECOOsum0 + ECOOsum1 + ECOOsum2 + ECOOsum3 + ECOOsum4 + ECOOsum5 + ECOOsumsub1  
  
 # 这里是收入计量  
  
 ECOIsum0 = 0  
 for y in range(0, len(getECOI\_0)):  
 ECOIsum0 += getECOI\_0[y].ProjectGot  
  
 ECOIsum1 = 0  
 for y1 in range(0, len(getECOI\_1)):  
 ECOIsum1 += getECOI\_1[y1].ProjectGot  
  
 ECOIsumsub1 = 0  
 for y2 in range(0, len(getECOI\_sub1)):  
 ECOIsumsub1 += getECOI\_sub1[y2].ProjectGot  
  
 ECOIsum = ECOIsum0 + ECOIsum1 + ECOIsumsub1  
  
 # 获取收入支出总值  
 newdata['ECOOsum'] = ECOOsum  
 newdata['ECOIsum'] = ECOIsum  
 # 获取支出每类总值  
 newdata['ECOOsum0'] = ECOOsum0  
 newdata['ECOOsum1'] = ECOOsum1  
 newdata['ECOOsum2'] = ECOOsum2  
 newdata['ECOOsum3'] = ECOOsum3  
 newdata['ECOOsum4'] = ECOOsum4  
 newdata['ECOOsum5'] = ECOOsum5  
 newdata['ECOOsumsub1'] = ECOOsumsub1  
 # 获取收入每类总值  
 newdata['ECOIsum0'] = ECOIsum0  
 newdata['ECOIsum1'] = ECOIsum1  
 newdata['ECOIsumsub1'] = ECOIsumsub1  
 # 取值完成  
 ################################  
 # print('总支出：' + str(ECOOsum) + '总收入：' + str(  
 # ECOIsum) + '支出小类分别：' + str(ECOOsum0) + ',' + str(ECOOsum1) + ',' + str(ECOOsum2) + ',' + str(  
 # ECOOsum3) + ',' + str(ECOOsum4) + ',' + str(  
 # ECOOsum5) + ',' + str(ECOOsumsub1) + ',' + '收入小类分别：' + str(ECOIsum0) + ',' + str(ECOIsum1) + ',' + str(  
 # ECOIsumsub1))  
 return HttpResponse(json.dumps(newdata))  
  
  
# model\_to\_dict只能实行一个QuerySet的转换，  
  
@csrf\_exempt  
def PDInfoUpdate(request):  
 PDAllObj = ProductionList.objects.all()  
  
 newdataList = []  
  
 for k in range(0, len(PDAllObj)):  
 dataUnit = []  
 dataUnit.append(PDAllObj[k].ProductionNM)  
 dataUnit.append(PDAllObj[k].ProductionUnitPrice)  
 dataUnit.append(PDAllObj[k].ProductionSoldNum)  
 dataUnit.append(PDAllObj[k].ProductionPrice)  
 dataUnit.append(PDAllObj[k].ProductionSoldDate)  
 # print(dataUnit)  
 newdataList.append(dataUnit)  
  
 # 添加safe=False,获取JsonRep  
 # print(newdataList)  
 # 获取日期索引的方法  
 def takeDate(elem):  
 return elem[4]  
  
 # 获取总价  
 def takeTotalPriceData(elem):  
 return elem[3]  
  
 # 消除非本月数据，清洗数据，并根据总售价排序数据  
 def cleanOtherMouthData(DataList):  
 thisYearDate = datetime.datetime.now().year  
 getDataLength = len(DataList)  
 getnewdataList = []  
 for i in range(0, getDataLength):  
 if str(thisYearDate) in DataList[i][4]:  
 getnewdataList.append(DataList[i])  
  
 # print('获取到的本年数据：' + str(getnewdataList))  
 getnewdataList.sort(key=takeTotalPriceData, reverse=True)  
 # print('获取到的根据价格排序的数据：' + str(getnewdataList))  
 return getnewdataList  
  
 newdataList = cleanOtherMouthData(newdataList)  
  
 # 获取总价前十的数据的方法  
 def getTop10Data(DataList):  
 getnewdataList = []  
 for i in range(0, 10):  
 getnewdataList.append(DataList[i])  
  
 return getnewdataList  
  
 # 获取到总价前十数据，将其从小到大排序  
 newdataList = getTop10Data(newdataList)  
 newdataList.sort(key=takeTotalPriceData)  
  
 # print('这是实际获取到的根据价格排序的数据：' + str(newdataList))  
  
 return HttpResponse(json.dumps(newdataList))  
  
  
@csrf\_exempt  
def PDFormUpdate(request):  
 newdata = {  
 'PDName': request.POST.get('PDName'),  
 'UnitPrice': request.POST.get('UnitPrice'),  
 'SoldNum': request.POST.get('SoldNum'),  
 'TotalPrice': request.POST.get('TotalPrice'),  
 'SoldDate': request.POST.get('SoldDate')  
 }  
 try:  
 PDGetObj = ProductionList.objects.get(ProductionNM=newdata['PDName'], ProductionSoldDate=newdata['SoldDate'])  
 except ProductionList.DoesNotExist:  
 print('这是一个新的产品信息。')  
 newdata['Message'] = '这是一个新的产品信息,已创建新项目！'  
 PDCreateObj = ProductionList(ProductionNM=newdata['PDName'], ProductionUnitPrice=newdata['UnitPrice'],  
 ProductionSoldNum=newdata['SoldNum'],  
 ProductionPrice=newdata['TotalPrice'],  
 ProductionSoldDate=newdata['SoldDate'])  
 PDCreateObj.save()  
  
 else:  
 print('这是一个已经存在的产品信息。')  
 newdata['Message'] = '这是一个已经存在的产品信息,已更新项目！'  
 PDGetObj.ProductionNM = newdata['PDName']  
 PDGetObj.ProductionUnitPrice = newdata['UnitPrice']  
 PDGetObj.ProductionSoldNum = newdata['SoldNum']  
 PDGetObj.ProductionPrice = newdata['TotalPrice']  
 PDGetObj.ProductionSoldDate = newdata['SoldDate']  
 PDGetObj.save()  
  
 return HttpResponse(json.dumps(newdata))  
  
  
# 此方法用于管理员发送通知所使用的AJAX，关联‘@func1’  
@csrf\_exempt  
def InfoUpdate(request):  
 newdata = {}  
 newdata['TextContent'] = request.POST.get('TextContent')  
 newdata['writername'] = request.POST.get('writername')  
 newdata['writedate'] = request.POST.get('writedate')  
 # print(newdata)  
 try:  
 getInfoData = InfoList.objects.get(TextContent=newdata['TextContent'], writername=newdata['writername'],  
 writedate=newdata['writedate'])  
 except InfoList.DoesNotExist:  
 # 获取本人是否发送过同样的消息  
 getInfoData = InfoList.objects.filter(TextContent=newdata['TextContent'], writername=newdata['writername'])  
 if len(getInfoData) == 0:  
 getInfoData = InfoList.objects.filter(TextContent=newdata['TextContent'])  
 if len(getInfoData) == 0:  
 # print("我们没有查询到同样信息。")  
 newdata['Message'] = "我们没有查询到通知过这条信息,所以我们同步发送了这条消息。"  
 createInfoData = InfoList(TextContent=newdata['TextContent'], writername=newdata['writername'],  
 writedate=newdata['writedate'])  
 createInfoData.save()  
 else:  
 # print("我们查询到在有人发送过同样信息")  
 newdata['NewDataConfirmTag'] = 1  
 # print(str(getInfoData[0]))  
 if len(getInfoData) > 1:  
 newdata['Message'] = "我们查询到有人(例如:" + getInfoData[len(getInfoData) - 1].writername + \  
 ")发送过同样信息,但是不一定是最近发的，所以建议您确认并更新本次操作，我们会将您的数据更新同步发送出去。您确认本次操作吗？"  
 elif len(getInfoData) == 1:  
 newdata['Message'] = "我们查询到有人(例如:" + str(getInfoData[0].writername) + \  
 ")发送过同样信息,但是不一定是最近发的，所以建议您确认并更新本次操作，我们会将您的数据更新同步发送出去。您确认本次操作吗？"  
  
 # 此处传回，并在接受到用户操作后跳转至新方法  
  
 elif len(getInfoData) > 1:  
 getSameDate = getInfoData[len(getInfoData) - 1].writedate  
 newgetInfoData = getInfoData[len(getInfoData) - 1]  
 newgetInfoData.writedate = newdata['writedate']  
 newgetInfoData.save()  
 # print("我们查询到您在[" + getSameDate + "]发送过同样信息,所以我们更新了您的发布时间并同步发送了这条消息。")  
 newdata['Message'] = "我们查询到您在[" + getSameDate + "]发送过同样信息,所以我们更新了您的发布时间并同步发送了这条消息。"  
 elif len(getInfoData) == 1:  
 # print(getInfoData[0].writedate)  
 getSameDate = getInfoData[0].writedate  
 getInfoData[0].writedate = newdata['writedate']  
 getInfoData[0].save()  
 # print("我们查询到您在[" + getSameDate + "]发送过同样信息,所以我们更新了您的发布时间并同步发送了这条消息。")  
 newdata['Message'] = "我们查询到您在[" + getSameDate + "]发送过同样信息,所以我们更新了您的发布时间并同步发送了这条消息。"  
  
 else:  
 # print("我们查询到在同一时间您发送的信息，数据库不更新！")  
 newdata['Message'] = "我们查询到您在同一时间发送过此信息。"  
  
 return HttpResponse(json.dumps(newdata))  
  
  
# 此方法用作当用户确认后才执行的二次Ajax操作,属于‘@func1’  
@csrf\_exempt  
def SureInfoDataUpdate(request):  
 newdata = {}  
 newdata['TextContent'] = request.POST.get('TextContent')  
 newdata['writername'] = request.POST.get('writername')  
 newdata['writedate'] = request.POST.get('writedate')  
 createInfoData = InfoList(TextContent=newdata['TextContent'], writername=newdata['writername'],  
 writedate=newdata['writedate'])  
 createInfoData.save()  
 newdata['Message'] = '已同步更新！'  
 return HttpResponse(json.dumps(newdata))  
  
  
@csrf\_exempt  
def showmenewInfo(request):  
 newdata = []  
 getOBj = InfoList.objects.all()  
  
 def getFormatTodayDate():  
 year = str(datetime.datetime.now().year)  
 mouth = '0' + str(datetime.datetime.now().month) if datetime.datetime.now().month < 10 else \  
 str(datetime.datetime.now().month)  
 day = '0' + str(datetime.datetime.now().day) if datetime.datetime.now().day < 10 else \  
 str(datetime.datetime.now().day)  
 return year + '-' + mouth + '-' + day  
  
 def getFormatYesterdayDate():  
 year = str(datetime.datetime.now().year)  
 mouth = '0' + str(datetime.datetime.now().month) if datetime.datetime.now().month < 10 else \  
 str(datetime.datetime.now().month)  
 day = '0' + str(datetime.datetime.now().day - 1) if datetime.datetime.now().day - 1 < 10 else \  
 str(datetime.datetime.now().day - 1)  
 return year + '-' + mouth + '-' + day  
  
 yesterdayDate = getFormatYesterdayDate()  
 todayDate = getFormatTodayDate()  
 print(yesterdayDate + ' ' + todayDate)  
 for i in range(0, len(getOBj)):  
 data = []  
 print(getOBj[i].writedate)  
 if yesterdayDate in getOBj[i].writedate or todayDate in getOBj[i].writedate:  
 data.append(getOBj[i].TextContent)  
 data.append(getOBj[i].writername)  
 data.append(getOBj[i].writedate)  
 newdata.append(data)  
  
 def gettime(elem):  
 return elem[2]  
  
 newdata.sort(key=gettime)  
 # newdata['Message'] = '我们显示了最新的20条不同管理员的消息，其它消息'  
 print(newdata)  
 return HttpResponse(json.dumps(newdata))  
# 获取人员权限  
@csrf\_exempt  
def getAuthority(request):  
 newdata = {}  
 newdata['username'] = request.POST.get('username')  
 try:  
 getOBJ = administratorNameList.objects.get(adminname=newdata['username'])  
 except administratorNameList.DoesNotExist:  
 newdata['adminFlag'] = -1  
 newdata['Message'] = '您是普通用户，不可以发送消息通知！'  
 else:  
 if getOBJ.adminPVLevel == 1:  
 newdata['adminFlag'] = 1  
 newdata['Message'] = '尊敬的超管，您可以使用任何功能！'  
 elif getOBJ.adminPVLevel == 2:  
 newdata['adminFlag'] = 2  
 newdata['Message'] = '您是管理员，可以发送消息！'  
 else:  
 newdata['adminFlag'] = 0  
 newdata['Message'] = '您的行为是非法的！'  
  
 return HttpResponse(json.dumps(newdata))  
  
  
# 获取收入项目，预测用  
@csrf\_exempt  
def getFutureInfo(request):  
 newdata = []  
 getECOIall = COMECIList.objects.all()  
 for i in range(0, len(getECOIall)):  
 datanode = []  
 # 生成随机数因子  
 datanode.append(random.random())  
 datanode.append(getECOIall[i].ProjectGot)  
 newdata.append(datanode)  
  
 return HttpResponse(json.dumps(newdata))

附录1.3 项目MyJsList.js文件（三种重要算法）

//金额转换算法  
function formatNum(num) {  
 // 将数字串转换成带逗号的显示方式  
 if (!/^([+\-])?(\d+)(\.\d+)?$/.test(num)) {  
 console.log("wrong!");  
 return num;  
 }  
 let a = RegExp.$1;  
 let b = RegExp.$2;  
 let c = RegExp.$3;  
 let re = new RegExp().compile("(\\d)(\\d{3})(,|$)");  
 while (re.test(b))  
 b = b.replace(re, "$1,$2$3");  
 return a + "" + b + "" + c;  
}  
//消除特殊字符算法  
function removeTip(str) {  
 while (true) {  
 if (str.indexOf('+') == -1 && str.indexOf('-') == -1 && str.indexOf(',') == -1) {  
 break;  
 } else {  
 str = str.replace('+', '');  
 str = str.replace('-', '');  
 str = str.replace(',', '');  
 }  
  
 }  
 return str;  
}  
  
//获得标准格式的日期  
function getdate(date) {  
 var year = date.getFullYear();  
 var month =  
 date.getMonth() + 1 < 10 ?  
 "0" + (date.getMonth() + 1) :  
 date.getMonth() + 1;  
 var day =  
 date.getDate() < 10 ? "0" + date.getDate() : date.getDate();  
 var hours =  
 date.getHours() < 10 ? "0" + date.getHours() : date.getHours();  
 var minutes =  
 date.getMinutes() < 10 ?  
 "0" + date.getMinutes() :  
 date.getMinutes();  
 var seconds =  
 date.getSeconds() < 10 ?  
 "0" + date.getSeconds() :  
 date.getSeconds();  
 let time =  
 year +  
 "-" +  
 month +  
 "-" +  
 day +  
 " " +  
 hours +  
 ":" +  
 minutes +  
 ":" +  
 seconds;  
 return time  
}

附录1.4 项目页面文件

<!DOCTYPE html>  
<html lang="en">  
  
<body>  
<nav class="navbar navbar-dark bg-dark">  
 <a class="navbar-brand" href="">  
 <img src="/static/JHOri.png" height="30px" width="30px" class="d-inline-block align-top" alt="error">  
 <strong class="wordfontchange"><span>简颢科技管理系统</span></strong>  
 </a>  
 <div class="linkpart">  
 <div class="welcomelabel">欢迎，</div>  
 <div class="usernamelabel">{{ username }}</div>  
 <div class="welcomelabel"> ！</div>  
 </div>  
</nav>  
<div class="box">  
 <div class="rowbox" id="rowbox">  
 <a href="javascript:;" id="memberDataTake">人员数据收集</a>  
 <a href="javascript:;" id="ComECO">公司财产事务</a>  
 <a href="javascript:;" id="marketECO">市场盈亏数据</a>  
 <a href="javascript:;" id="VTData">数据分析及可视化</a>  
 <a href="javascript:;" id="infoPlate">内部消息通报</a>  
 </div>  
 <div class="contentbox clearfix" id="contentbox">  
 {# <div>username:{{ username }}</div>#}  
 {# <div>password:{{ password }}</div>#}  
 {# <div>LoginState:{{ errorUorP }}</div>#}  
 {# <div><a href="../">back</a></div>#}  
 {# 背景模块 /90/B#}  
 <div class="FirstPage">  
 <div>JH TECHNOLOGY CO LTD</div>  
 </div>  
 {# 人员数据收集/90/B#}  
 <div index="memberDataTake" id="memberDataTakePart" class="memberDataTakePart">  
 <div>人员数据收集</div>  
 {# 人员数据收集表单#}  
 <form action="/memInfoList/" method="post" class="memberInfoForm">  
 {% csrf\_token %}  
 <div class="clearfix">  
 <label for="nameTextPlace" class="wordfontchange">姓名:</label>  
 <input type="text" placeholder="请输入姓名" name="nameTextPlace" id="nameTextPlace" class="form-control"  
 maxlength="20"  
 style="width: 15%;text-align: center;">  
 <label for="nationTextPlace" class="wordfontchange">民族:</label>  
 <label for="NPTextPlace" class="wordfontchange">籍贯:</label>  
 <input type="text" placeholder="请输入籍贯" name="NPTextPlace" id="NPTextPlace" class="form-control"  
 maxlength="8"  
 style="width: 15%;text-align: center;">  
 <label for="brithPlaceTextPlace" class="wordfontchange">出生地:</label>  
 <input type="text" placeholder="请输入出生地" name="brithPlaceTextPlace" id="brithPlaceTextPlace"  
 class="form-control" maxlength="8"  
 style="width: 15%;text-align: center;">  
  
 </div>  
 <div class="secendPart clearfix">  
 <label class="wordfontchange">性别:</label>  
 <input type="radio" id="sexmaleChoosePlace" name="sexChoose" checked="checked" value="male">  
 <label class="wordfontchange" for="sexmaleChoosePlace">male</label>  
 <input type="radio" id="sexfemaleChoosePlace" name="sexChoose" value="female">  
 <label class="wordfontchange" for="sexfemaleChoosePlace">female</label>  
 <label for="brithTextPlace" class="wordfontchange" style="margin-left: 22px;">出生年月:</label>  
 <input type="month" id="brithTextPlace" name="brithTextPlace" min="1930-01" value="1930-01"  
 style="margin-top: 4px;">  
 <label for="HealthTextPlace" class="wordfontchange">健康状况:</label>  
 <input type="text" placeholder="默认填（健康）" id="HealthTextPlace" name="HealthTextPlace"  
 class="form-control" maxlength="8"  
 style="width: 15%;text-align: center;margin-top: 0;">  
 <label for="TelTextPlace" class="wordfontchange">联系电话:</label>  
 <input type="text" placeholder="请输入联系电话" name="TelTextPlace" id="TelTextPlace" class="form-control"  
 maxlength="20"  
 style="width: 20%;text-align: center;margin-top: 0;">  
 </div>  
 <div class="thirdPart clearfix">  
 <label for="emailTextPlace" class="wordfontchange">个人邮箱:</label>  
 <input type="email" placeholder="sophie@example.com" id="emailTextPlace" name="emailTextPlace"  
 class="form-control"  
 maxlength="100"  
 style="width: 35%;text-align: center;margin-top: 0;">  
 <label for="EDTextPlace" class="wordfontchange">学历:</label>  
 <input type="text" placeholder="|大专|本科|硕士|博士|" id="EDTextPlace" name="EDTextPlace"  
 class="form-control"  
 maxlength="2"  
 style="width: 20%;text-align: center;margin-top: 0;">  
 </div>  
 <div class="forth clearfix">  
 <label for="expTextPlace" class="wordfontchange">个人过往（工作、实习经历）:</label>  
 <div>  
 <textarea rows="15" cols="100" placeholder="过往经历:年月-年月 经历" id="expTextPlace"  
 name="expTextPlace"  
 class="form-control"  
 style="width: 95%;margin: 0 auto;float: none;"></textarea>  
 </div>  
  
 </div>  
 <div class="btnline" style="text-align: right;margin: 5px 25px;">  
 <button type="reset" class="btn btn-primary" style="margin-right: 4px;" id="resetBtn">reset</button>  
 <script>  
 var textAreaPlace = document.getElementById('expTextPlace');  
 var resetBtn = document.getElementById('resetBtn');  
 resetBtn.onclick = function () {  
 textAreaPlace.innerText = "";  
 }  
 </script>  
 <button type="button" class="btn btn-success" id="submitBtn">submit</button>  
 {# 根据session中meminfodataexisted的值判定是否成功上传#}  
 <script>  
 $('#submitBtn').click(function () {  
 $.ajax({  
 url: '/memInfoList/',  
 type: 'post',  
 data: {  
 'username': $('#nameTextPlace').val(),  
 'nation': $('#nationTextPlace').val(),  
 'NP': $('#NPTextPlace').val(),  
 'birthplace': $('#brithPlaceTextPlace').val(),  
 'sex': $('input[name="sexChoose"]:checked').val(),  
 'birthday': $('#brithTextPlace').val(),  
 'health': $('#HealthTextPlace').val(),  
 'tel': $('#TelTextPlace').val(),  
 'email': $('#emailTextPlace').val(),  
 'education': $('#EDTextPlace').val(),  
 'expText': $('#expTextPlace').val(),  
 },  
 success: function (data) {  
 {#console.log(data)#}  
 var newdata = JSON.parse(data)  
 alert(newdata['Message'])  
 }  
 })  
 });  
 </script>  
 {#var getMemListBtn = document.getElementById('submitBtn');#}  
 {#getMemListBtn.addEventListener('mouseover', function () {#}  
 {# console.log('i active');#}  
 {# console.log(sessionStorage.getItem('meminfodataexisted'));#}  
  
  
 </div>  
  
 </form>  
  
 </div>  
 {# 公司收益模块/90/B#}  
 <div index="ComECO" id="ComECOpart" class="ComECOpart">  
 <div class="label">公司收支</div>  
 <div class="content clearfix">  
 <div>  
 <div id="label0">自研项目</div>  
 <div>  
 <div id="EIDiv0">  
 <div>  
 <label>收入:</label>  
 <div id="EIDataText0"></div>  
 </div>  
 </div>  
 <div id="EODiv0">  
 <div>  
 <label>支出:</label>  
 <div id="EODataText0"></div>  
 </div>  
 </div>  
 </div>  
 </div>  
 <div>  
 <div id="label1">外方合作</div>  
 <div>  
 <div id="EIDiv1">  
 <div>  
 <label>收入:</label>  
 <div id="EIDataText1"></div>  
 </div>  
 </div>  
 <div id="EODiv1">  
 <div>  
 <label>支出:</label>  
 <div id="EODataText1"></div>  
 </div>  
 </div>  
 </div>  
 </div>  
 <div>  
 <div id="label2">人员工资</div>  
 <div>  
 <div id="EODiv2">  
 <div>  
 <label>支出:</label>  
 <div id="EODataText2"></div>  
 </div>  
 </div>  
 </div>  
 </div>  
 <div>  
 <div id="label3">税务支出</div>  
 <div>  
 <div id="EODiv3">  
 <div>  
 <label>支出:</label>  
 <div id="EODataText3"></div>  
 </div>  
 </div>  
 </div>  
 </div>  
 <div>  
 <div id="label4">宣传支出</div>  
 <div>  
 <div id="EODiv4">  
 <div>  
 <label>支出:</label>  
 <div id="EODataText4"></div>  
 </div>  
 </div>  
 </div>  
 </div>  
 <div>  
 <div id="label5">员工开销</div>  
 <div>  
 <div id="EODiv5">  
 <div>  
 <label>支出:</label>  
 <div id="EODataText5"></div>  
 </div>  
 </div>  
 </div>  
 </div>  
 {# 支出项目提交#}  
 <div>  
 <div id="label6">支出项目提交</div>  
 <div>  
 <div id="EODiv6" style="margin: 15px auto;">  
 <div>  
 <label>总支出:</label>  
 <div id="totalODiv"></div>  
 </div>  
 </div>  
 <div>  
 <div style="text-align: left">  
 {# 支出提交表单#}  
 <form action="javascript:;" method="post">  
 {% csrf\_token %}  
 <label class="DiyLabel">项目明细（名称）:</label>  
 <div>  
 <input type="text" maxlength="50" class="form-control" id="ECOOProjectName"  
 style="height: 25px; width: 90%; margin: 5px auto;">  
 </div>  
 <label class="DiyLabel">项目支出:</label>  
 <div style="display: inline-block;">  
 <input type="text" maxlength="20" class="form-control" id="ECOOProjectCost"  
 style="height: 25px; width: 125px; margin: 5px auto;">  
 </div>  
 <div style="text-align: right;">  
 <button type="reset" class="btn btn-secondary"  
 style="margin-right: 5px; height: 25px;line-height: 25px;padding-top: 0;">  
 reset  
 </button>  
 <button type="button" class="btn btn-success" id="ECOOBtn"  
 style="height: 25px;line-height: 25px; padding-top: 0;margin-right: 10px;">  
 submit  
 </button>  
 </div>  
 </form>  
 {# 支出表单提交AJAX#}  
 <script>  
 $("#ECOOBtn").click(function () {  
 if ($('#ECOOProjectName').val() == "" && $('#ECOOProjectCost').val() == "") {  
 alert('您还没填写内容！');  
 } else {  
 var confirmFlag = confirm("你确定要添加这个新项目吗？\n项目名称:" +  
 $('#ECOOProjectName').val() + "\n项目支出:" + $('#ECOOProjectCost').val())  
 if (confirmFlag) {  
 ECOOProFormUpdata();  
 }  
 }  
 })  
  
 //提交支出项目表单的方法  
 function ECOOProFormUpdata() {  
 $.ajax({  
 url: '/ECOOList/',  
 type: 'post',  
 data: {  
 'projectNM': $('#ECOOProjectName').val(),  
 'projectCost': $('#ECOOProjectCost').val(),  
 },  
 success: function (data) {  
 var writedata = JSON.parse(data)  
 alert(writedata['Message'])  
 }  
 })  
 }  
 </script>  
 </div>  
 </div>  
 </div>  
 </div>  
 {# 收入项目提交#}  
 <div>  
 <div id="label7">收入项目提交</div>  
 <div>  
 <div id="EIDiv2" style="margin: 15px auto;">  
 <div>  
 <label>总收入:</label>  
 <div id="totalIDiv"></div>  
 </div>  
 </div>  
 <div>  
 <div style="text-align: left">  
 {# 收入提交表单#}  
 <form action="">  
 {% csrf\_token %}  
 <label class="DiyLabel">项目明细（名称）:</label>  
 <div>  
 <input type="text" maxlength="50" class="form-control" id="ECOIProjectName"  
 style="height: 25px; width: 90%; margin: 5px auto;">  
 </div>  
 <label class="DiyLabel">项目收入:</label>  
 <div style="display: inline-block;">  
 <input type="text" maxlength="20" class="form-control" id="ECOIProjectGot"  
 style="height: 25px; width: 125px; margin: 5px auto;">  
 </div>  
 <div style="text-align: right;">  
 <button type="reset" class="btn btn-secondary"  
 style="margin-right: 5px; height: 25px;line-height: 25px;padding-top: 0;">  
 reset  
 </button>  
 <button type="button" class="btn btn-success" id="ECOIBtn"  
 style="height: 25px;line-height: 25px; padding-top: 0;margin-right: 10px;">  
 submit  
 </button>  
 </div>  
 </form>  
 {# 支出表单提交AJAX#}  
 <script>  
 $("#ECOIBtn").click(function () {  
 if ($('#ECOIProjectName').val() == "" && $('#ECOIProjectGot').val() == "") {  
 alert('您还没填写内容！');  
 } else {  
 var confirmFlag = confirm("你确定要添加这个新项目吗？\n项目名称:" +  
 $('#ECOIProjectName').val() + "\n项目收入:" + $('#ECOIProjectGot').val())  
 if (confirmFlag) {  
 ECOIProFormUpdata();  
 }  
 }  
 })  
  
 //提交收入项目表单的方法  
 function ECOIProFormUpdata() {  
 $.ajax({  
 url: '/ECOIList/',  
 type: 'post',  
 data: {  
 'ProjectNM': $('#ECOIProjectName').val(),  
 'ProjectGot': $('#ECOIProjectGot').val(),  
 },  
 success: function (data) {  
 var writedata = JSON.parse(data)  
 alert(writedata['Message'])  
 }  
 })  
 }  
 </script>  
 </div>  
 </div>  
 </div>  
 </div>  
 {# 标题头、收入、支出相关样式脚本#}  
 <script>  
 {#设置标题头样式、收入div、支出div的CSS#}  
 var getLabel = [];  
 //批量使用标题头样式  
 for (var i = 0; i < 8; i++) {  
 getLabel[i] = document.getElementById('label' + i);  
 }  
 var LabelObj = getLabel;  
 for (var i = 0; i < LabelObj.length; i++) {  
 getLabel[i].className = 'comEClabel';  
 }  
 //设置收入div样式  
 var getEIdiv = [];  
 for (var i = 0; i < 3; i++) {  
 getEIdiv[i] = document.getElementById('EIDiv' + i);  
  
 }  
 var EIObj = getEIdiv;  
 for (var i = 0; i < EIObj.length; i++) {  
 if (i >= 0 && i < 2) {  
 EIObj[i].className = 'EIDivCSS';  
 } else {  
 EIObj[i].className = 'EIDivCSS2';  
 }  
 }  
 //设置盒子下面元素的样式表  
 for (var i = 0; i < EIObj.length; i++) {  
 EIObj[i].querySelector('div').className = 'clearfix';  
 EIObj[i].querySelector('label').className = 'EIDivLabel';  
 EIObj[i].querySelector('div').querySelector('div').className = 'EIDivData';  
 }  
 //设置支出模块样式  
 var getEOdiv = [];  
 for (var i = 0; i < 7; i++) {  
 getEOdiv[i] = document.getElementById('EODiv' + i);  
 }  
 var EOObj = getEOdiv;  
 for (var i = 0; i < EOObj.length; i++) {  
 if (i < 2 && i >= 0) {  
 EOObj[i].className = 'EODivCSS';  
 } else {  
 EOObj[i].className = 'EODivCSS2';  
 }  
  
 }  
 //console.dir(EIObj);  
 //设置盒子下面元素的样式表  
 for (var i = 0; i < EOObj.length; i++) {  
 EOObj[i].querySelector('div').className = 'clearfix';  
 EOObj[i].querySelector('label').className = 'EODivLabel';  
 EOObj[i].querySelector('div').querySelector('div').className = 'EODivData';  
 }  
 </script>  
  
 {# 实现支出和收入的动态更新#}  
 {# id="totalODiv"、id="totalIDiv"总支出，总收入#}  
 {# 2个收入标签、6个支出标签，[0~2)，[0~6)#}  
 {# EODataText+K支出标签，EIDataText+K收入标签#}  
  
 <script>  
  
  
 {#设置全局时钟，监听全局事件，伴随页面关闭#}  
 {############################################}  
 {#1.设置ECO动态刷新时钟变量#}  
 {#获取ECOIO小类和总和收入及支出AJAX方法#}  
  
 function getECOIO() {  
 $.ajax({  
 url: '/getECOIO/',  
 type: 'post',  
 data: {},  
 success: function (data) {  
 var newdata = JSON.parse(data)  
 {#总支出和总收入#}  
 var strECOOsum = formatNum(newdata['ECOOsum'])  
 var strECOIsum = formatNum(newdata['ECOIsum'])  
 {#支出分类#}  
 var strECOOsumClassDetail = {  
 'strECOOsum0': formatNum(newdata['ECOOsum0']),  
 'strECOOsum1': formatNum(newdata['ECOOsum1']),  
 'strECOOsum2': formatNum(newdata['ECOOsum2']),  
 'strECOOsum3': formatNum(newdata['ECOOsum3']),  
 'strECOOsum4': formatNum(newdata['ECOOsum4']),  
 'strECOOsum5': formatNum(newdata['ECOOsum5']),  
 'strECOOsumsub1': formatNum(newdata['ECOOsumsub1']),  
 }  
 {#收入分类#}  
 var strECOIsumClassDetail = {  
 'strECOIsum0': formatNum(newdata['ECOIsum0']),  
 'strECOIsum1': formatNum(newdata['ECOIsum1']),  
 'strECOIsumsub1': formatNum(newdata['ECOIsumsub1']),  
 }  
  
 {#设置总收入和总支出显示#}  
 $('#totalODiv').text('-' + strECOOsum)  
 $("#totalIDiv").text('+' + strECOIsum)  
 {#设置分类支出显示#}  
 for (let i = 0; i < 6; i++) {  
 $('#EODataText' + i).text('-' + strECOOsumClassDetail['strECOOsum' + i])  
 }  
 {#设置分类收入显示#}  
 for (let i = 0; i < 2; i++) {  
 $('#EIDataText' + i).text('+' + strECOIsumClassDetail['strECOIsum' + i])  
 }  
 }  
 })  
 }  
  
 {#设置ECO#}  
 var timerDisplayECOData = undefined;  
 {#ECO生存事件阶段模块#}  
 {#设定开关标记，防止无限生成时钟#}  
 var flag = 0;  
 {#2.设置ECO动态刷新时钟、动态函数#}  
 var globalTimer = setInterval(function () {  
 var displayState = $('#contentbox #ComECOpart').css('display');  
 if (displayState == 'block') {  
 {#console.log('test:' + timerDisplayECOData);#}  
 if (flag == 0) {  
 timerDisplayECOData = setInterval(getECOIO, 1000);  
 flag = 1  
 }  
 } else if (displayState == 'none') {  
 if (timerDisplayECOData != undefined) {  
 clearInterval(timerDisplayECOData);  
 flag = 0  
 }  
 }  
 }, 500)  
 </script>  
 </div>  
 </div>  
 {# 市场盈利模块/90/B#}  
 <div index="marketECO" id="marketECOpart" class="marketECOpart">  
 <div class="label">市场盈利</div>  
 <div class="content clearfix">  
 <div id="bigBtn" class="bigBtn" style="background-color: rgba(215,241,191,0.35);">  
 <div class="ChartLabel">今年市场销售额前10名（月计）</div>  
 <div id="ercharts1" style="width: 100%;height: 90%;"></div>  
  
 </div>  
 <div id="marketInfoForm" class="bigBtn" style="background-color: rgba(186,183,183,0.35);">  
 <form action="javascript:;" method="post" class="Productionform clearfix">  
 <label for="PDNameText">ProductionName:</label>  
 <input type="text" class="form-control" id="PDNameText" style="width: 80%;">  
 <label for="UnitPrice">UnitPrice:</label>  
 <input type="number" class="form-control" id="UnitPriceText" style="width: 20%;">  
 <label for="SoldNumText">SoldNum:</label>  
 <input type="number" class="form-control" id="SoldNumText" style="width: 20%;">  
 <label for="TotalPriceText">TotalPrice:</label>  
 <input type="text" class="form-control" id="TotalPriceText" style="width: 20%;"  
 disabled="disabled" value="" placeholder="该项目会自动计算！">  
 {#自动获取输入内容，并自动计算总价脚本#}  
 <script>  
 //自动获取输入内容，并自动计算总价  
 function AutoField() {  
 let UnitPrice = $('#UnitPriceText').val()  
 let SoldNum = $('#SoldNumText').val()  
  
 if (UnitPrice != null && SoldNum != null) {  
 $('#TotalPriceText').val((UnitPrice \* SoldNum).toString());  
 }  
 }  
  
 $('#UnitPriceText').bind('input', AutoField)  
 $('#SoldNumText').bind('input', AutoField)  
 </script>  
 <label for="">SoldDate:</label>  
 <input type="month" class="form-control" id="SoldDateText" style="width: 20%;">  
 <div class="btnPlace">  
 <button type="reset" class="btn btn-warning">Reset</button>  
 <button type="button" class="btn btn-success" id="updataPDInfoBtn">更新项目订单</button>  
 </div>  
 </form>  
 <script>  
 $('#updataPDInfoBtn').click(function () {  
 if ($('#PDNameText').val() != '' && $('#UnitPriceText').val() != '' &&  
 $('#SoldNumText').val() != '' && $('#SoldDateText').val() != '') {  
 var getSure = confirm("您确定要更此内容吗？\n项目名字:" + $('#PDNameText').val() + ' 单价:' +  
 $('#UnitPriceText').val() + '\n当月销售量:' + $('#SoldNumText').val() + ' 总销售额:' +  
 $('#TotalPriceText').val() + '\n销售日期:' + $('#SoldDateText').val())  
 if (getSure) {  
 {#更新产品表项至数据库#}  
 $.ajax({  
 url: '/PDFormUpdate/',  
 type: 'post',  
 data: {  
 'PDName': $('#PDNameText').val(),  
 'UnitPrice': $('#UnitPriceText').val(),  
 'SoldNum': $('#SoldNumText').val(),  
 'TotalPrice': $('#TotalPriceText').val(),  
 'SoldDate': $('#SoldDateText').val()  
 },  
 success: function (data) {  
 {#console.log(data);#}  
 var newdata = JSON.parse(data);  
 alert(newdata['Message']);  
 }  
 })  
 }  
 } else {  
 alert('请输入内容，不要留空！')  
 }  
 })  
 </script>  
 </div>  
 <div id="UpdataPDErchart" class="UpdataPDErchartBtn">  
 <button type="button" class="btn btn-success" id="UPDErchartBtn">生成新表</button>  
 </div>  
 {#图标绘制脚本#}  
   
 </div>  
 {# 重新绘制支出#}  
 <div>  
 <label for="redrawTheOdepartECBtn">重新绘制支出占比图:</label>  
 <button type="button" id="redrawTheOdepartECBtn" class="btn btn-dark">重新绘制</button>  
 </div>  
  
 <div>  
 <label for="redrawTheFuturnECBtn">重新绘制收入推演图:</label>  
 <button type="button" id="redrawTheFuturnECBtn" class="btn btn-dark">重新推演</button>  
 <script>  
 function showercharts6(getFuturedata) {  
 var chartDom = document.getElementById('erchart6');  
 var myChart = echarts.init(chartDom);  
 var option;  
  
 //这里是预测的显示和刷新脚本  
  
 $.ajax({  
 url: '/getFutureInfo/',  
 type: 'post',  
 data: {},  
 success: function (data) {  
 var newdata = JSON.parse(data)  
 {#收入数据回收#}  
 {#console.log(newdata)#}  
 showercharts6(newdata);  
 }  
 })  
  
 $('#redrawTheFuturnECBtn').click(function () {  
 $.ajax({  
 url: '/getFutureInfo/',  
 type: 'post',  
 data: {},  
 success: function (data) {  
 var newdata = JSON.parse(data)  
 {#收入数据回收#}  
 var getercharts6 = document.getElementById('erchart6');  
 getercharts6.parentElement.removeChild(getercharts6.parentElement.children  
 [getercharts6.parentElement.children.length - 1])  
 var geterchart6div = document.getElementById('ercharts6DivPlace');  
 var newercharts6Node = document.createElement('div');  
 newercharts6Node.className = 'erchartsplace';  
 newercharts6Node.id = 'erchart6';  
 geterchart6div.appendChild(newercharts6Node)  
 showercharts6(newdata);  
 }  
 })  
 })  
 </script>  
 </div>  
 </div>  
  
 </div>  
  
 </div>  
 </div>  
 {# 通知模块#}  
 <div index="infoPlate" id="infoPlatepart" class="infoPlatepart">  
 <div class="label">通知</div>  
 <div class="content">  
 <div class="infoPart">  
 <div class="Textcontent">  
 <div id="InfoPlace" style="height:100%;overflow-y:auto">  
  
 </div>  
 </div>  
 </div>  
 <div class="departLine"></div>  
 <div class="sendPlace clearfix">  
 <div class="sendInfoPlace">  
 <textarea name="inputInfoarea" id="inputInfoarea"  
 class="form-control inputInfoarea"></textarea>  
 </div>  
 <div class=" sendInfoBtnPlace">  
 <button id="allclearBtn" class="btn btn-info">清除所有消息</button>  
 <button id="showNewInfoBtn" class="btn btn-success">ShowMeInfo</button>  
 <button id="sendInfoBtn" class="btn btn-success">SendMessage</button>  
 </div>  
 </div>  
  
 {# 清楚所有消息的脚本 #}  
 <script>  
 $('#allclearBtn').click(function () {  
 $('#InfoPlace').children().remove();  
 })  
 </script>  
  
 {# 显示最新的消息通知按钮的脚本#}  
  
 {# 这是发送消息按钮的脚本#}  
 <script>  
 //这里设置了管理员按发送按钮的方法  
 //判定权限次数位  
 var PvTag = 0;  
 var getSendBtn = document.getElementById('sendInfoBtn');  
  
 function UpdataInfoAction() {  
 //实现数据库同步消息内容  
 $.ajax({  
 url: '/InfoUpdate/',  
 type: 'post',  
 data: {  
 'TextContent': getInputInfoArea.value,  
 'writername': '{{ username }}',  
 'writedate': date  
 },  
 success: function (data) {  
 var newData = JSON.parse(data)  
 if (newData['NewDataConfirmTag'] != 1) {  
  
 alert(newData['Message']);  
 } else {  
  
 var IsSure = confirm(newData['Message']);  
 if (IsSure) {  
 $.ajax({  
 url: '/SureInfoDataUpdate/',  
 type: 'post',  
 data: newData,  
 success: function (data) {  
 let newData = JSON.parse(data)  
 alert(newData['Message'])  
 }  
 })  
 } else {  
 alert('已取消同步！')  
 }  
 }  
  
  
 }  
 })  
  
 }  
 }  
  
 getSendBtn.onclick = function () {  
 $.ajax({  
 url: '/getAuthority/',  
 type: 'post',  
 data: {  
 'username': '{{ username }}',  
 },  
 success: function (data) {  
 var newdata = JSON.parse(data)  
  
  
 if (newdata['adminFlag'] == 0) {  
 $(document.body).attr("disabled", "disabled");  
 alert(newdata['Message'])  
  
 } else if (newdata['adminFlag'] == -1) {  
 $('#sendInfoBtn').attr("disabled", "disabled");  
 $('#inputInfoarea').attr("disabled", "disabled");  
 alert(newdata['Message'])  
  
 } else if (newdata['adminFlag'] == 1) {  
 if (PvTag == 0) {  
 PvTag = 1;  
 alert(newdata['Message']);  
 UpdataInfoAction();  
 } else {  
 UpdataInfoAction();  
 }  
  
  
 } else if (newdata['adminFlag'] == 2) {  
 if (PvTag == 0) {  
 PvTag = 1;  
 alert(newdata['Message']);  
 UpdataInfoAction();  
 } else {  
 UpdataInfoAction();  
 }  
 }  
 }  
 })  
  
 }  
  
 //这里我们判定权限是否可以进行编写发送功能  
  
  
 </script>  
 </div>  
  
 </div>  
  
 </div>  
 <script>  
 var rowbox = document.getElementById('rowbox');  
 var contentBox = document.getElementById('contentbox');  
 var actionChoice = rowbox.getElementsByTagName('a');  
 var contentDivPart = contentBox.children;  
  
 for (var i = 0; i < contentDivPart.length; i++) {  
 if (i != 0) {  
 contentDivPart[i].style.float = "left";  
 contentDivPart[i].style.display = 'none';  
 }  
 }  
 for (var i = 0; i < actionChoice.length; i++) {  
 actionChoice[i].onmouseover = function () {  
 var getIdIndex = this.getAttribute('id');  
 var contentDiv = contentBox.children;  
 for (var j = 0; j < contentDiv.length; j++) {  
 contentDiv[j].style.display = 'none';  
 }  
 for (var j = 0; j < contentDiv.length; j++) {  
 var getIndex = contentDiv[j].getAttribute('index');  
 if (getIdIndex == getIndex) {  
 contentDiv[j].removeAttribute('style');  
 {#contentDiv[j].removeAttribute('class');#}  
 contentDiv[j].style.display = 'block';  
 }  
 }  
  
  
 }  
 }  
  
 {#for (var i = 0; i < ; i++) {#}  
  
 {#actionChoice.onmouseover = function () {#}  
 {# console.log(actionChoice);#}  
  
  
 </script>  
</div>  
</body>  
</html>