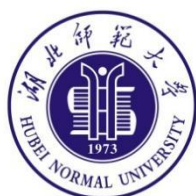


分类号 TP391

学号 2020Z08541420

学校代码 10513

密级 公开



湖北师范大学
HUBEI NORMAL UNIVERSITY

硕士学位论文

(专业学位)

基于 Spring Boot 和 Vue 框架的企业绩效考

核系统设计与实现

作者姓名：胡金宇

专业学位类别：电子信息

专业学位领域：电子信息

指导教师：江国舟 副教授

答辩日期：2023 年 5 月

**A Thesis Submitted in Fulfillment of Requirements for the
Degree of Master of Electronics information**

**Design and implementation of enterprise
performance appraisal system based on
Spring Boot and vue framework**

Candidate : Jinyu Hu

Major : Electronic Information

Supervisor : Guozhou Jiang

Hubei Normal University

Huangshi 435002, P. R. China

May ,2023

摘 要

随着社会经济的高速发展和市场竞争的加剧,绩效考核对企业管理的作用愈发重要。传统的绩效考核采用填写考核报表等方式,存在报表数量多、工作效率低下和人工统计分析误差等问题。这些问题导致绩效考核失去原有的意义,不仅浪费了企业的时间和金钱,还未能为企业的竞争力和综合水平增添任何价值。因此,开发和设计一套符合公司实际需求、科学的绩效考核系统至关重要。该系统能提高企业自身的竞争力,促进员工的职业发展,并提高绩效评价的精准度和效率。

本文首先对绩效考核的基础概念、原则以及考核方式进行了深入分析,并介绍了多种成熟的考核方式,如 360 度绩效考核法、关键绩效指标考核法(Key Performance Indicator, KPI)以及平衡计分卡考核法(Balanced Score Card, BSC)。根据系统业务需求和非功能性需求,本文建立了企业绩效考核指标,并对系统的总体架构、各模块以及数据库进行了设计。然后,本文根据计算机软件工程思想并结合 KPI 关键绩效指标考核法,开发了一套采用前后端分离的企业绩效考核系统。该系统后端采用了 Spring Boot 框架,整合了 Mybatis 和 Redis 等技术,使用 MVC 模式(Model-View-Controller)构建系统总体架构;前端采取 Vue 框架并整合了 Vuex 和 vue-router 等技术,采用 Element-UI 设计前端页面,基于 MVVM (Model-View-ViewModel)开发设计模式,构建单页面应用(single page application, SPA)。根据企业绩效考核系统的设计与需求,着重阐述了系统管理模块和绩效考核模块的实现过程。最后为了保证企业绩效考核系统的稳定性,分别对系统进行了功能测试和非功能测试。测试结果验证了系统可以在高强度的环境下正常运行。

本文实现的企业绩效考核系统采用模块化开发,各模块之间耦合度低,内聚性强,有助于后期系统的升级和优化。该系统正式上线后,可进一步提高员工的工作效率,对企业管理和员工绩效的提升具有积极意义,符合现代信息技术的发展趋势。

关键词: 绩效考核; 信息系统; Spring Boot; Vue

Abstract

With the rapid development of social economy and the intensification of market competition, the role of performance appraisal is becoming more and more important for enterprise management. The traditional performance appraisal adopts such methods as filling out appraisal reports, which has problems such as a large number of reports, low efficiency and errors in manual statistical analysis. These problems cause performance appraisal to lose its original meaning, which not only wastes the time and money of the company, but also fails to add any value to the competitiveness and comprehensive level of the company. Therefore, it is crucial to develop and design a scientific performance appraisal system that meets the actual needs of the company. This system can improve the competitiveness of the company itself, promote the career development of employees, and improve the accuracy and efficiency of performance evaluation.

This paper firstly analyzes the basic concepts, principles and assessment methods of performance appraisal in depth, and introduces a variety of mature assessment methods, such as 360-degree performance appraisal method, Key Performance Indicator (KPI) appraisal method and Balanced Score Card (BSC) appraisal method. According to the system business requirements and non-functional requirements, this paper establishes the enterprise performance appraisal index and designs the overall architecture, modules and database of the system. Then, based on the computer software engineering idea and combined with the KPI key performance indicator assessment method, this paper developed a set of enterprise performance appraisal system using front and back-end separation. The back-end of the system adopts Spring Boot framework, integrates Mybatis and Redis technologies, and uses MVC pattern (Model-View-Controller) to build the overall architecture of the system; the front-end adopts Vue framework and integrates Vuex and vue-router technologies, and adopts Element-UI to design the front-end page. The front-end adopts the Vue

framework and integrates technologies such as Vuex and vue-router, and adopts Element-UI to design the front-end pages, and builds a single page application (SPA) based on the MVVM (Model-View-ViewModel) development design pattern. According to the design and requirements of the enterprise performance appraisal system, the implementation process of the system management module and performance appraisal module is highlighted. Finally, in order to ensure the stability of the enterprise performance appraisal system, functional and non-functional tests are conducted for the system respectively. The test results verified that the system can operate normally under a high-intensity environment.

The enterprise performance appraisal system implemented in this paper adopts modular development, with low coupling between modules and strong cohesion, which helps upgrade and optimize the system later. After the system is officially launched, it can further improve the work efficiency of employees, which is of positive significance to enterprise management and employee performance improvement and is in line with the development trend of modern information technology.

Key words: Performance appraisal; Information System; Spring Boot; Vue

目 录

| | |
|----------------------------|----|
| 1 绪论..... | 1 |
| 1.1 课题背景与意义..... | 1 |
| 1.2 国内外研究现状..... | 2 |
| 1.2.1 国外研究现状..... | 2 |
| 1.2.2 国内研究现状..... | 3 |
| 1.3 论文主要工作..... | 4 |
| 1.4 论文组织结构..... | 5 |
| 2 绩效考核理论与关键技术 | 6 |
| 2.1 绩效考核基础理论..... | 6 |
| 2.1.1 绩效考核的概念..... | 6 |
| 2.1.2 绩效考核的原则..... | 6 |
| 2.1.3 绩效考核的方式..... | 7 |
| 2.2 关键技术分析..... | 7 |
| 2.2.1 Spring Boot 框架 | 7 |
| 2.2.2 Vue 框架 | 9 |
| 2.2.3 其他相关技术介绍..... | 11 |
| 2.2.4 前后端分离的意义..... | 12 |
| 2.3 本章小结..... | 12 |
| 3 系统需求分析 | 13 |
| 3.1 系统建设目标..... | 13 |
| 3.2 业务需求分析..... | 14 |
| 3.3 功能性需求分析..... | 16 |
| 3.3.1 系统管理需求..... | 16 |
| 3.3.2 考核评估需求..... | 20 |
| 3.4 非功能性需求分析..... | 23 |
| 3.5 本章小结..... | 24 |
| 4 绩效考核系统的设计 | 25 |
| 4.1 绩效考核指标设计..... | 25 |
| 4.2 总体架构设计..... | 27 |
| 4.3 系统功能模块设计..... | 29 |
| 4.4 系统数据库设计..... | 30 |
| 4.4.1 概念结构设计..... | 30 |
| 4.4.2 逻辑结构设计..... | 31 |
| 4.4.3 物理结构设计..... | 32 |

| | |
|-------------------|----|
| 4.5 本章小结..... | 36 |
| 5 系统的实现与测试 | 37 |
| 5.1 系统环境部署..... | 37 |
| 5.2 系统模块实现..... | 38 |
| 5.2.1 系统登录模块..... | 38 |
| 5.2.2 人员管理模块..... | 39 |
| 5.2.3 部门管理模块..... | 41 |
| 5.2.4 岗位管理模块..... | 42 |
| 5.2.5 绩效管理模块..... | 42 |
| 5.2.6 绩效考核模块..... | 44 |
| 5.3 系统测试..... | 45 |
| 5.3.1 系统测试方法..... | 45 |
| 5.3.2 功能性测试..... | 45 |
| 5.3.3 非功能性测试..... | 47 |
| 5.4 本章小结..... | 47 |
| 6 总结与展望..... | 48 |
| 6.1 全文总结..... | 48 |
| 6.2 工作展望..... | 49 |
| 参考文献..... | 50 |

1 绪论

1.1 课题背景与意义

在当今信息技术日益成熟的时代，企业与信息技术联系变得日益紧密。随着企业为应对激烈的竞争而不断创新，对员工工作效率的要求也不断提高。绩效考核系统将企业绩效管理与计算技术和信息化相结合，相对于传统的单一绩效考核方式，它为企业的绩效考核工作提供了便利和系统化的平台，实现了全员参与，进而提高了企业绩效管理和考核的效率和准确性^[1]。采用系统化的方式进行企业的绩效管理工作，能够大大缩短工作时间、节约成本，并减轻人力资源部门的工作负担。这不仅有助于企业的发展，还可以激发员工的工作热情，提高工作效率，实现双赢的效果。

然而本课题所研究的企业仍存在许多问题，例如采用传统的 Excel 报表进行绩效考核，这导致相关部门提交考评数据的滞后。同时，该考核流程需要进行自评表单的统计和结果收集，这导致工作难度加剧，并且存在错报、漏报的情况。此外，整个流程也缺乏计算机系统的数据支持。实际上，该企业已经开发了多款内部管理平台，具备内部开发能力和外部网络设备的支持，因此可以开发一套企业绩效考核系统来解决这些问题。

该企业绩效考核系统基于软件工程设计思想和系统开发流程编写，参考了大量优秀参考文献和实际调研数据，以确保系统的科学性和实用性，该系统的设计和实现对企业发展具有重要意义。该系统首先可以提高企业信息共享的能力，使各部门之间的沟通更畅通、协作更高效；其次，它可以提高人力资源管理的效率，有助于企业更好地管理和发展员工。这将帮助企业更好地吸引和留住人才，提高员工的工作满意度和忠诚度，从而为企业的可持续发展奠定坚实基础。

绩效考核系统可以实时显示和更新员工的绩效考核信息。员工可以随时登录绩效考核系统查看自己的绩效考核情况，以明确工作目标，提升自身的工作质量、实现薪资与工作质量的对应，促进自身职业生涯的良好发展；另一方面，绩效考核系统可以减少以往手工考核的时间和经济成本，解决 Excel 表格的数据收集和信息统计耗时等问题。

本文研究了企业绩效考核系统的设计与实现，在系统选型时采用了如今主流

的前后端技术，实现了对绩效考核理论知识与系统设计实现深入探究。研究成果能够提高企业绩效考核管理工作的效率，提升绩效考核的客观性和公正性。

1.2 国内外研究现状

绩效考核国内外研究内容主要是包括绩效考核理论、体系和绩效考核信息化平台的建设、发展两方面。

1.2.1 国外研究现状

绩效考核是一种起源于西方发达国家的企业人力资源管理方法。1813 年美国军方就引入了绩效考核模式。1824 年美国联邦政府开始对公务员进行绩效考核评估。Campbell 等学者在国外的绩效管理研究工作中提出了“绩效是员工控制自身与组织目标相关的行为”的观点被广泛的认同和赞赏^[2]。然而，不同国家的绩效评估体系因地区文化和思想差异而存在差异。美国将职位分析纳入绩效考核，使得绩效管理与企业发展战略相一致。而在欧洲地区，绩效考核起源于工业革命，主要考核对象是工人，考核目的是确保员工与岗位的匹配，并对员工薪酬进行管理；而日韩等亚洲国家的绩效考核更加注重企业的收益，主要以企业部门为考核对象^[3]。国外长期的绩效考核发展催生了多种成熟的考核方法。目标管理法(Management By Objective, MBO)、关键绩效指标法(Key Performance Indicator, KPI)、平衡计分卡(Balanced Score Card, BSC)、360 度绩效考核法以及关键事件法(Critical Incident Method, CIM)等^[4]都是比较常见的绩效考核方法。这些方法通过明确目标、量化绩效、建立绩效指标、评估绩效并提供反馈等手段，帮助企业或组织更好地管理和提升绩效。

除了绩效考核理论的发展外，国外早已建设了实用的绩效考核信息化平台。凭借着计算机与网络技术的较早发展，一些西方国家已开始使用绩效考核系统进行员工考核。然而，当时互联网技术的不成熟导致绩效考核信息无法共享等问题，使得绩效考核管理存在一定困难。随着计算机和软件技术的飞速发展，绩效考核信息的互通性早已被解决，国外知名企业的绩效考核系统大部分已经实现了信息平台的建设意义。例如：甲骨文、微软、SAP 等建设的企业绩效管理系统。甲骨文推出了企业资源计划(Enterprise Resource Planning, ERP) 产品，该产品可以根据客户的需求进行高度定制化，以适应各种行业和企业的需求。该系统集成

了财务管理、采购管理、供应链管理、销售管理、生产管理、人力资源管理等多个方面。该系统采用 Java 语言开发，具有跨平台性、安全性高。该系统可以与 Office 和与 Excel 等软件进行数据交换。此外，该系统还提供多种应用程序编程接口(Application Programming Interface,API)，供用户与其他应用系统进行交互实现数据共享和集成^[5]，加强了该系统的可扩展性。SAP 公司的 ERP 系统采用模块化设计，涵盖了企业的财务、采购、销售、人力资源和品质监管等多个方面，可应用于零售、公共事业、媒体以及工厂等多个行业。该系统具有完备的管理思想和高稳定性，能快速处理大量的实时数据，提供数据分析和决策支持，支持全球化业务操作，包括跨境采购、多货币结算等，但由于该系统开发语言为 SAP 公司特有的 ABAP 语言，因此与第三方系统的集成度较低^[6]。

1.2.2 国内研究现状

我国在绩效考核管理方面的研究起步较晚，但国外成熟的绩效考核体系为我国研究提供了宝贵的经验。2001 年，清华大学的张鼎昆教授对人类绩效技术进行了专门研究后强调了绩效考核技术的实践重要性，并认为绩效技术主要理论基础是系统论、行为科学、认知科学、神经科学和人力资源管理。2004 年，郭娟等人提出，企业在绩效考核评估过程中也可以引入平衡记分卡方法，而不仅仅是政府可以采用该方法^[7]。2006 年，吕华在其著作《当前企业绩效管理中存在的问题和对策》中指出我国企业绩效考核存在以下问题：绩效考核管理观念落后、考核指标设计不科学、考核模式存在主观性等。他主张企业应以战略为导向，构建科学的绩效管理体系。2010 年蒋春丽指出企业考核的核心应以人为本，不能仅考虑硬性数据如考核指标。她强调绩效考核应建立激励体系，在制定绩效目标前及实施绩效任务期间应与员工进行沟通，并且在考核后要向员工进行反馈^[8]。以上学者和教授对绩效考核研究做出了杰出贡献，但也表明了我国企业绩效考核存在的问题和挑战。

20 世纪 90 年代，随着国内计算机和信息化技术的迅速发展，大部分的企业和银行都开始将人力资源管理的绩效考核工作应用到信息化平台中，例如金蝶公司和用友公司开发的企业绩效管理系统^{[9][10]}。2017 年，李顺章开发了一套针对电力企业的绩效考核系统，该系统引入平衡记分卡的考核方式，将企业的战略发展以内部业务流程、财务、客户、学习和成长四个维度。该考核系统覆盖企业的

管理工作，管理层通过使用该系统提供的数据来管理和客观实施策略。该企业的绩效考核系统是基于 WEB 访问模式的员工绩效考核系统，它具有非常丰富的核心内容，可以真实反映评价对象的绩效^[11]。目前，国内的绩效考核系统少数系统采用 C/S 模式，大多数系统选择 B/S 模式，开发语言包括 java、VS.Net，相关模型有 MVC、SSH 框架等，常用数据库包括 Oracle、SQL Server 或 MySQL 等。

尽管国内在绩效考核理论和实践方面已经取得了一定的进展，但我国的绩效管理仍然面临一些问题和挑战。一些企业在实践中缺乏有效的绩效考核指标和方法，这导致考核结果的准确性和可靠性受到影响；一些企业过于注重量化指标和数据，在绩效评估中忽视了员工的主观能动性和非量化方面的表现，导致绩效管理效果不佳。由于不同企业的文化和组织结构存在较大差异，因此需要量身定制绩效考核方案，这对于绩效管理人员的素质和能力提出了更高的要求。

1.3 论文主要工作

本论文以绩效考核相关理论为基础，结合该公司实际业务需求以及当前流行的系统开发技术，为企业设计开发一款具有实用性和可靠性的系统。该绩效考核系统将取代传统的人工填写自评表的考核模式，提供公正、公平的考核结果，调动员工积极性。具体工作包括以下内容：

（1）安装与调试系统的运行软硬件环境，介绍系统开发所涉及的绩效考核理论、和绩效考核系统开发所用的相关技术，包括前后端框架和数据库等。

（2）进行需求分析。基于企业绩效考核要求的分析，将制定详细的系统要求，其中包括系统建设目标、业务需求、功能和非功能需求。根据这些要求设计企业绩效考核系统。

（3）系统详细设计。设计系统整体架构，将系统分为多个功能模块。

（4）系统的实现。包括编写和调试各个功能模块的代码，确保实现所有系统功能。遵循代码规范，注重代码的可读性、可维护性和可扩展性。

（5）系统测试。对系统进行全面的测试和分析，包括功能测试和非功能性测试。

1.4 论文组织结构

第一章，绪论。主要介绍本论文研究背景与意义、国内外绩效考核系统的发展现状以及研究内容。

第二章，绩效考核理论和关键技术。主要介绍了绩效考核概率、原则和成熟的绩效考核方法，同时介绍了绩效考核系统中运用的关键技术。

第三章，系统需求分析。本章根据企业实际绩效考核现状进行需求分析，得出系统的建设目标、业务需求、功能性需求和非功能性需求。

第四章，绩效考核系统的设计。本章基于绩效考核系统的需求分析完成了绩效考核系统的详细设计，包括绩效考核体系设计、系统的总体架构设计、系统功能模块设计、系统数据库设计。

第五章，系统的实现和测试。本章根据绩效考核系统的详细设计，对绩效考核系统的系统管理模块和考核评估模块进行实现。最后，对系统的功能测试和性能测试进行分析，得出测试结论。

第六章，总结与展望。总结本课题研究设计实现的企业绩效考核系统，并在现有的工作基础上，提出进一步优化工作的展望。

2 绩效考核理论与关键技术

本章主要介绍了有关绩效考核基础理论和企业绩效考核系统开发中采用的关键技术。首先，对绩效考核理论进行简要分析，为设计科学的企业绩效考核体系提供基础。随后，介绍了系统开发过程中所用到的技术和涉及的相关研究。

2.1 绩效考核基础理论

2.1.1 绩效考核的概念

绩效考核是企业人力资源管理的重要组成部分，对企业管理中发挥着重要作用。它对员工的工作表现、产出结果和成就进行考核，并向被考核者提供反馈。绩效考核包括一系列过程，例如设定考核内容，实施考核计划，进行考核，并对考核结果进行反馈^[12]。绩效考核的目的是评估员工的工作表现、工作态度、工作质量，以便为企业提供更好的管理和决策依据，调整工作岗位和薪酬，降低运营成本，实现效益最大化，同时也可以帮助员工了解自身的优劣势并提高工作能力和职业发展。

2.1.2 绩效考核的原则

企业绩效考核的通用性原则包括客观公正、公开透明和及时反馈等。由于各企业对绩效考核的要求各异，因此需要针对企业自身特点，制定科学的绩效考核原则，如下所示：

（1）目标导向性原则：绩效考核应与企业整体方向和长期计划一致，确保考核结果能够对企业目标的实现产生积极的推动作用。

（2）激励性原则：绩效考核结果应该与员工的薪酬、晋升、职称评定等激励制度相关联，以激发员工的积极性。

（3）综合性原则：绩效考核应该综合考虑员工在多个方面的表现，如工作业绩、能力素质、工作态度、团队协作等，避免片面评价。

（4）差异化原则：对于公司内不同部门，应该根据其业务特定进行差异性考核，以确保考核结果公平公正。

2.1.3 绩效考核的方式

随着越来越多的政府部门、企业和高校对绩效考核的重视，其基本理论体系不断创新发展，绩效考核方法也更加完善。绩效考核主要应用于评估被考核对象的业务能力、工作量、工作质量以及工作业绩。目前已经出现了一些常见且较为成熟的绩效考核方法，包括 360 度绩效考核法、关键绩效指标考核法(KPI)以及平衡计分卡考核法(BSC)等。具体如下所述：

(1) 360 度绩效考核法：该方法是一种全面的企业绩效考核方法，它通过对员工进行多维度评价，如自我评价、同级评价、上级评价、下属评价等。这种方法对员工的表现进行了全方位评价和反馈，使其适合于对中层及以上员工进行评估，为企业选拔人才、建立正确导向提供支持。然而，360 度绩效考核法也存在缺点，包括综合考核过程耗时长、成本高^[13]，同时全面公正的考核结果也会受到信息、认知、情绪层面等因素的影响。

(2) 关键绩效指标考核法（KPI）：关键绩效指标法是业内广泛使用的一种绩效考核方法^[14]。KPI 考核法的实施过程通常包括以下几个步骤：首先，确定关键绩效指标并量化；其次，制定考核计划和标准，明确考核对象、考核时间和考核标准；然后，收集数据并进行分析和衡量，得出员工的绩效得分；最后，对绩效得分进行总结和反馈，制定绩效改进计划，不断完善绩效考核体系。这种方法将员工的个人目标与组织的战略目标联系起来。

(3) 平衡计分卡考核法（BSC）：平衡计分卡考核法是一种全面的绩效管理方法，它考虑了多个角度，包括财务、客户、内部流程、以及学习和成长。它使组织能够将其战略目标与绩效指标相结合，并衡量实现目标的进展^[15]。这种方法提供了一个平衡的组织绩效视图，并促进了不同部门之间的沟通和协作。然而，实施平衡计分卡可能很耗时，需要付出巨大的努力来开发和维护能够准确反映组织战略重点的指标。

2.2 关键技术分析

2.2.1 Spring Boot 框架

Spring Boot 是由 Pivotal 团队于 2013 年开发的框架，它在 Spring 框架的基础上进行了重新设计和开发，在继承 Spring 优秀的特性的同时改进其存在的不足

足之处。Spring Boot 通过提供一些默认配置，简化了 Spring 框架的使用，使得它更轻量化，让开发者能够更专注于业务逻辑的开发，进而提高开发效率。图 2-1 展示了 Spring Boot 应用程序的执行过程。

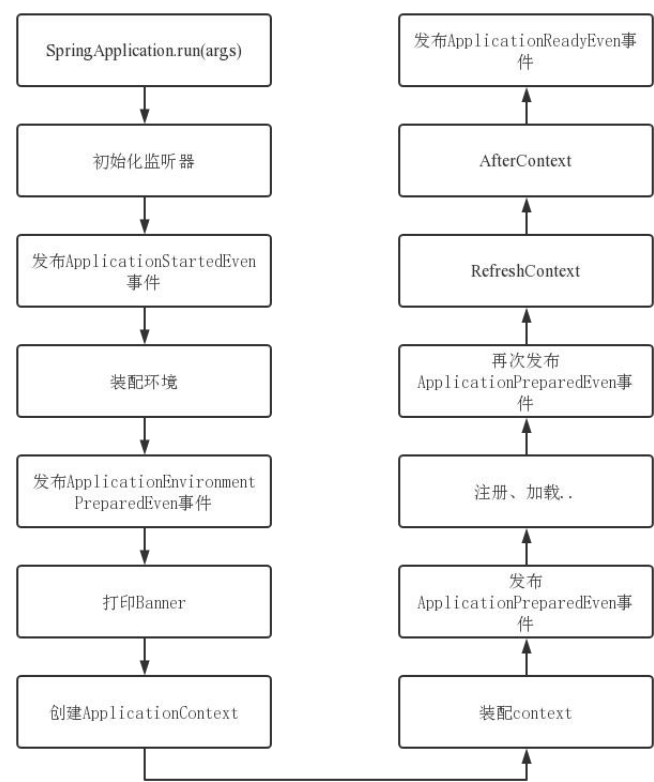


图 2-1 SpringApplication 执行流程

根据图 2-1 可知，SpringApplication 在完成参数初始化、配置加载和注册加载等过程后，会初始化一个监听器，并发布了 ApplicationReadyEvent 事件，用于表示 Spring Boot 项目已经启动完毕并准备好处理请求^[16]。

Spring Boot 是一个开源的 Java Web 框架，能够为开发人员提供更快、更简便的方式来构建基于 Spring 框架的 Web 应用程序。它采用自动化配置的方式，为开发者提供了一种简单的方式来快速构建、配置和部署应用程序，从而提高开发效率。

Spring Boot 的主要特点是自动化配置和微服务架构，使得开发人员不再需要花费大量时间进行繁琐的配置工作。它能够自动配置 Spring 应用程序，包括数据源、视图解析器、安全性等等。同时，Spring Boot 框架提供了丰富的第三

方库和插件，如 Hibernate、MyBatis、Spring Data 等，可以轻松地集成其他流行的框架和工具，从而大大简化了开发人员的工作。

Spring Boot 还提供了内嵌 Tomcat 等服务器的能力，使得应用程序能够快速部署和启动。此外，Spring Boot 还提供了一些非功能性特性，例如健康检查和外部配置，这些功能极大地提高了应用程序的可维护性和可扩展性。

在使用 Spring Boot 框架进行开发时，开发人员可以使用注解、JavaConfig 等方式来进行配置，从而使得代码更加简洁易懂，更加易于维护。

2.2.2 Vue 框架

在开发基于 B/S 架构的 Java Web 应用时，系统页界面的美观是系统的关键因素。通常情况下，前端界面的绘制和交互都是通过操作文档对象模型（DOM）完成的，但是这种方式容易报错^[17]。为了解决这一问题，前端框架一直在不断更新迭代，这些框架都是基于 MVVM（Model-View-ViewModel）设计模式构建的。MVVM 设计模式将视图、数据和业务逻辑分离，使得开发人员可以专注于业务逻辑的开发和维护。该设计模式的架构图如图 2-2 所示。MVVM 设计模式的优势使其成为了现代前端开发的主流设计模式，为前端项目的开发和维护带来了许多便利^[18]。

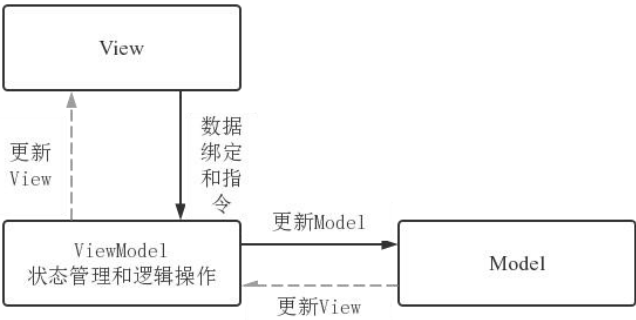


图 2-2 MVVM 设计模式

MVVM 设计模式是源于传统的 MVC 设计模式的一种模式，全称为 Model-View-ViewModel。该设计模式将系统分为三层：模型层、视图层和视图模型层。其中，模型层用于存储用户数据；视图层则负责呈现界面元素和控件；视图模型层将模型转换为可直接在视图上显示的数值。如图 2-2 所示，视图层将用

户的命令和绑定数据传递给视图模型层，后者会动态更新模型，当模型中的数据发生变化时，视图模型层会自动跟踪变化。该设计模式的优点在于解耦视图和模型之间的依赖关系，从而使得系统更易于开发、维护和扩展。

Vue.js 是当前最受欢迎的前端框架之一。Vue.js 的设计理念是提供一种简单易用的开发方式，同时保持高度的灵活性和可扩展性。Vue.js 采用了 MVVM（Model-View-ViewModel）架构模式^[19]。它能够将数据、视图和业务逻辑分离，从而提高前端开发的效率和简洁性。Vue.js 框架的原理如图 2-3 所示：

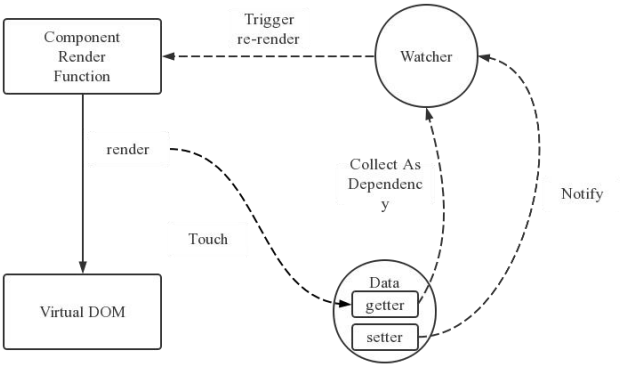


图 2-3 Vue 框架原理

Vue.js 采用 Virtual DOM 技术来优化渲染性能。Vue.js 将数据转换为 JavaScript 对象，创建一个 Virtual DOM 树，然后使用 diff 算法比较前后两个 Virtual DOM 树的差异，并最终只更新必要的 DOM 元素。使用 Virtual DOM 技术，避免了频繁操作 DOM 的性能问题，同时降低了代码的复杂度，使开发者不必手动操作 DOM 元素。这种响应式的机制免去了开发者手动维护数据和视图之间的关系，同时也使得应用的性能更加高效^[20]。

Vue.js 具备许多独特的特性，如数据绑定、组件化、路由和状态管理。其中，Vue.js 最引人注目的特性是数据双向绑定^[21]，这使得开发人员无需手动管理数据和 UI 的同步，从而大大降低了开发难度和出错率。此外，Vue.js 的组件化设计使得开发人员能够轻松地构建复杂的 UI 界面，并提高了代码的可复用性和可维护性。

Vue.js 框架的核心部分是 Vue 实例^[22]，每个 Vue 实例都是一个独立的作用域，拥有自己的数据、方法、计算属性和生命周期。通过 Vue 实例的绑定和事件监听，

实现数据的双向绑定，将数据和方法与 HTML 模板相互绑定。Vue.js 的指令系统也是一个重要的特性，例如 v-bind、v-if 和 v-for，这些指令使得开发人员能够方便地对 DOM 进行操作。

除了核心部分之外，Vue.js 还拥有许多插件和扩展，如 Vue Router、Vuex 和 Vue CLI 等。Vue Router 是 Vue.js 的官方路由插件，使得开发人员能够轻松实现页面之间的跳转和参数传递。Vuex 是 Vue.js 中的状态管理工具，主要用于组件间的状态共享和状态的变化跟踪。Vuex 将状态存储在一个名为 store 的全局存储器中，并提供了一组 API 用于更新和获取状态。它使得状态变化的过程可追溯、可预测和可调试。在大型的 Vue.js 应用程序中，使用 Vuex 可以提高应用程序的可维护性和开发效率^[23]。Vue CLI 是 Vue.js 的脚手架工具，能够快速创建一个 Vue.js 项目，并提供许多实用的工具和功能，例如代码打包、热重载、单元测试等。

2.2.3 其他相关技术介绍

（1）MyBatis 框架

MyBatis 是一种 Java 持久化框架，它能够将 Java 对象映射到数据库中的数据表中。MyBatis 的核心是 Mapper 接口和 XML 映射文件^[24]。通过这两个元素，可以完成对数据库的操作。使用 MyBatis 时，可以通过 Mapper 接口定义 SQL 语句，并使用 XML 映射文件将 SQL 语句与 Mapper 接口中的方法进行关联^[25]。MyBatis 具有高灵活性、易于扩展和优异的性能。

（2）Element-UI 框架

Element-UI 是一种基于 Vue.js 框架的前端 UI 组件库^[26]，它提供了丰富的基础组件和业务组件，包括按钮、表格、表单、菜单等。Element-UI 的设计风格简洁美观、易于使用，因此广泛应用于 Vue.js 项目中。同时，Element-UI 还提供了一套完整的主题定制方案，开发者可以通过定制主题来满足项目需求^[27]。

（3）Axios 技术

Axios 是一种基于 Promise 的 HTTP 客户端，它可以在浏览器和 Node.js 中使用^[28]。Axios 的优点在于易用性、拦截请求和响应、请求和响应的转换等等。Axios 支持 HTTP 请求和响应的拦截，因此可以在发送请求或接收响应前进行处理操作，例如设置请求头、请求超时时间、请求拦截器、响应拦截器等^[29]。

(4) MySQL 数据库与 Redis 数据库

MySQL 是一种开源的关系型数据库管理系统，支持多用户、多线程和多种存储引擎。MySQL 具有高性能、高可靠性和易用性等优点^[30]，在 Web 开发中广泛使用。MySQL 支持使用 SQL 语言进行数据的增删改查等操作。Redis 是一种高性能的非关系型数据库^[31]，它支持多种数据结构，包括字符串、列表、哈希、集合和有序集合。Redis 具有高性能、高可靠性、易用性等优点，在缓存、消息队列等场景中被广泛使用。Redis 的数据存储在内存中，因此读写速度很快。同时，Redis 还提供了发布/订阅、Lua 脚本、事务等多种功能。

2.2.4 前后端分离的意义

前后端分离是单页应用（SPA）开发中广泛采用的架构模式。网易云音乐网页版就是单页应用的优秀案例，它采用了部分刷新技术，避免了请求数据时的白屏等待时间。这种架构模式通过异步数据请求实现快速的数据加载，避免了部分数据变化时的卡顿，从而提高了用户操作的连续性和体验。此外，前端和后端工程相互独立，它们分别是两个独立的项目，这不仅提高了项目开发效率，还降低了项目维护和二次开发成本。

前后端分离的架构模式适用于职责划分明确的应用场景。在这种架构模式中，后端主要负责服务层的开发，从数据库中提取数据供前端使用，同时负责将前端返回的数据存储在数据库中以保证数据的可靠性^[32]。前端主要负责用户交互，绘制静态页面，并根据业务逻辑将数据呈现给用户。此外，前端还应该在后端开发中优化页面动态效果、响应速度、导航、操作一致性等^[33]。

前后端分离的意义主要在于更具体的职责划分，从而减少沟通成本^[34]；提高页面性能并减少内存消耗。此外，它还可以降低二次开发成本，让前端开发人员快速发现问题，提高组件重用率^[35]，从而快速推动项目开发。

2.3 本章小结

本章主阐述了该项目中涉及的绩效考核相关理论和关键技术。首先，介绍了绩效考核的基础概念概率、相关原则和常见的绩效考核方式。然后，描述了企业绩效考核系统中的关键技术，并说明了前后端分离对系统开发的重要性。

3 系统需求分析

需求分析是保证企业绩效考核系统质量的关键环节。本章将结合绩效考核理论和关键技术，了解企业对系统建设目标，结合实际的业务、功能性和非功能性需求，以获得该系统的详细需求。

3.1 系统建设目标

本次课题研究的企业采用纸质办公在公司内部实施绩效考核流程，各种工作单的批准和审核需要依靠人工传递纸质文件，这导致工作效率难以提高，同时带来额外的人工成本。现行的绩效考核模式存在以下问题：

（1）在进行绩效考核时，若考核者或审核人员由于各种原因不能到公司工作，将导致执行链中断，进而工作积压，降低了工作效率，绩效考核也无法发挥激励员工的作用。

（2）纸质文档流转的透明性相对较低，由于组织架构和其他因素，绩效考核过程中常常出现中断和受阻等现象。由于各种原因，当事人可能会搁置考核工作，并且没有专人负责追踪和催办，这导致无人知道文件停留在哪个环节，影响了绩效考核效率。

（3）整个绩效考核过程采用纸质文档，使查询和更新十分困难。在流转的过程中，文件可能会被损坏或丢失，这会导致前一环节需要重新实施，影响工作效率。此外，当需要总结考核成绩时，纸质文档虽然进行了分类保存，但仍然不便查找。

根据以上存在的问题，本系统的开发需要考虑以下几个方面：

（1）绩效考核指标体系的设计。本系统需要针对企业的 KPI 考核指标进行分析，设计更为科学优秀的考核指标，以确保考核结果的准确性和可信度。

（2）系统架构的设计与技术选型。本系统需要选择合适的技术和架构，以实现高可靠性的系统，并保证系统在考核期间能够稳定高效地运行。此外，本系统在线上完成考核流程，并采用各个模块流程提示，以提高用户体验和效率。

（3）通过绩效考核进行数据采集，为员工建立基本信息档案和绩效信息档案，以实现考核结果的自动化管理。

3.2 业务需求分析

鉴于企业可能存在更换组织架构的需求，因此在组织架构发生变化时，先前的部门绩效考核模板可能需要相应的修改。此外，还应该能够将更改后的部门与相应的绩效考核模板相关联。为了便于企业对其组织架构进行后期的维护，该系统给出了相关功能。本文以一家互联网企业为研究对象，对其组织架构进行了研究，如图 3-1 所示。该公司包括财务部、综合部、研发中心、营销中心和供应链中心。研发中心又包括测试部、硬件部和软件部。

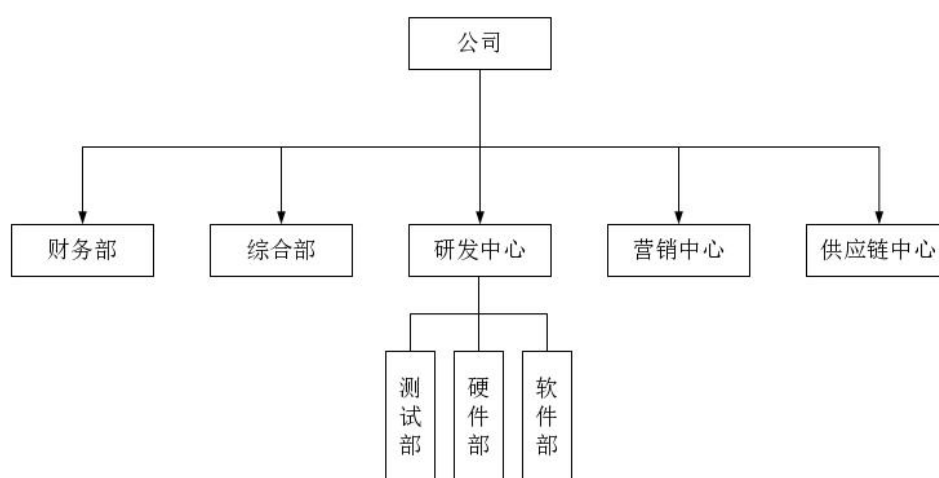


图 3-1 企业组织架构图

该系统的用户分为三种：管理员、高级员工和普通员工，按照权限级别递减顺序排序。不同类型的员工拥有不同的功能。普通员工可以系统登录、填写绩效考核自评表以及查询考核结果等。高级员工除了具有普通员工的功能外，还负责对普通员工绩效考核进行审批和流程管理。而管理员则拥有对企业绩效考核系统的信息设置权限。

(1) 普通员工：可以登录企业绩效考核系统，完成登录登出、修改密码、填写和提交绩效考核自评表，以及查看自己的绩效考核评价结果等操作。该角色通常指的是公司中的非管理岗员工。

(2) 高级员工：除了具备普通员工的功能外，还负责审核和评分普通员工的自评表，并查看所在部门内员工的考核成绩。该角色通常指公司各个部门的负责人和高级管理人员。

(3) 管理员：其职责包括维护企业绩效考核系统的稳定运行，例如维护绩效考核过程中人员、部门以及考核信息，以确保所有用户都可以正常使用该系统。此外，管理员还要定期备份和还原系统后台数据。通常由公司的信息部门员工担任，但管理员没有更改具体员工绩效考核评价数据信息的权限，以确保考核结果的公平公正。

企业绩效考核系统管理的总体流程如下图 3.2 所示。管理员首先发起企业绩效考核通知，制定相应部门考核模板。随后，普通员工开始填写自评表。只有员工填写好自评表并提交之后，高级员工才能对其进行打分。管理员会汇总综合成绩和排名，并导出给人力资源部进行存档。最终考核结果会反馈给普通员工，利用绩效考核结果直接与员工薪资水平挂钩，以达到有效的绩效考核。

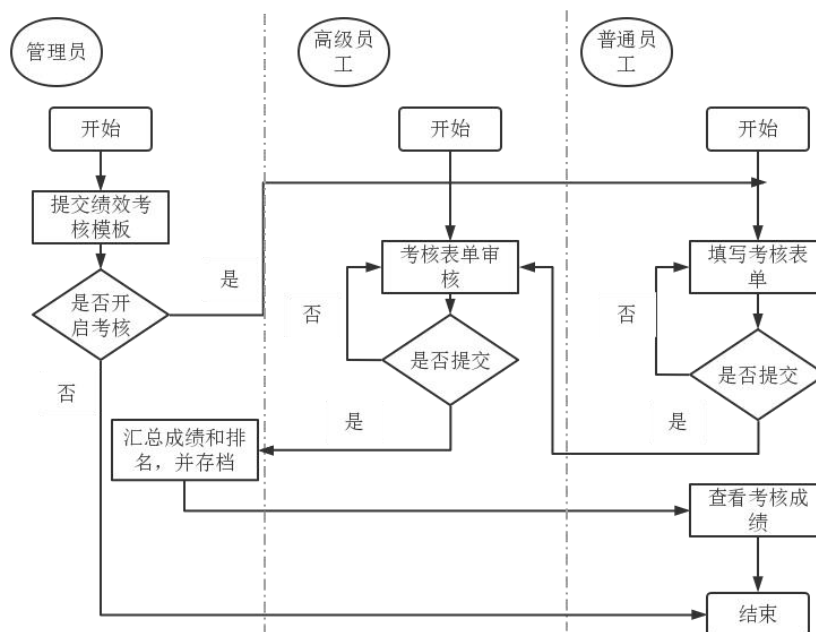


图 3-2 绩效管理总体流程图

3.3 功能性需求分析

系统功能性需求分析的主要目的是根据企业绩效考核系统的建成目标和业务需求，将其从整体功能细化为各个主要模块。通过结合企业绩效管理总体流程图，明确系统实际功能需求，并将其划分为系统管理模块和考核评估模块。系统管理模块包括登录校验、人员、部门和职业信息管理需求，考核评估模块包括考核信息和绩效考核管理需求。如图 3-3 所示为系统功能需求用例图。

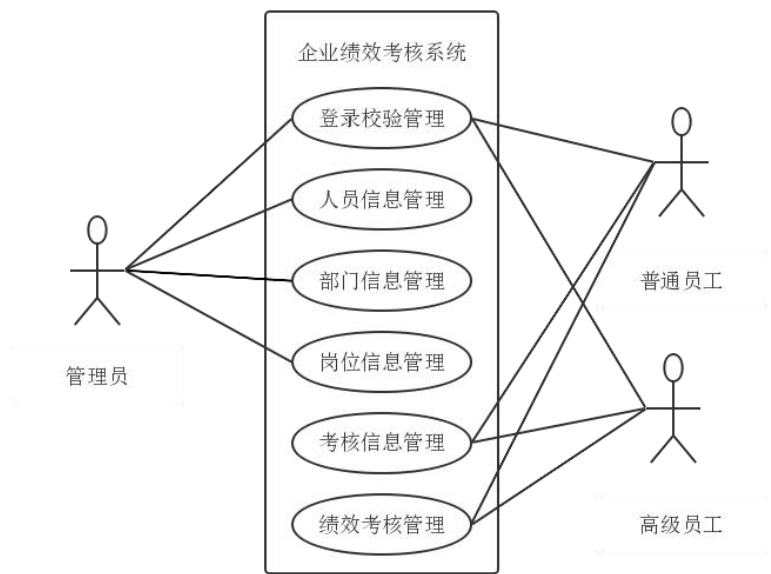


图 3-3 绩效考核系统用例图

3.3.1 系统管理需求

企业绩效考核系统管理需求主要用于对企业的基本信息维护管理，由多个功能模块组成。下面将详细介绍每个功能模块的具体设计。

(1) 系统登录模块

该模块是整个企业绩效考核系统的入口，用户进行绩效考核是通过输入用户名和密码登录绩效考核系统。当企业员工输入正确的账号和账号密码后，将成功进入系统，并获得自身的基本信息。企业绩效考核系统登录用例图如图 3-4 所示，用例功能描述表见表 3-1。

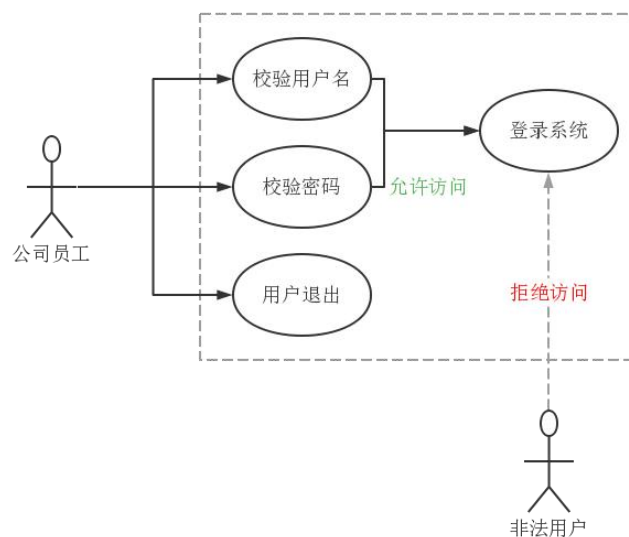


图 3-4 系统登录用例图

表 3-1 系统登录用例功能描述表

| 描述项 | 说明 |
|-------|--|
| 用例名称 | 企业绩效考核系统登录 |
| 参与者 | 企业绩效考核的合法用户 |
| 用例描述 | 该系统的合法用户通过输入账号和密码登录绩效考核系统。系统对输入的用户名和密码进行验证，如果用户名和密码正确，并且与数据库中的信息匹配，则用户被认为是合法用户。这样做可以保证系统的安全性和稳定性，只有经过授权的用户才能访问系统，从而确保系统的数据不会被非法访问或篡改。 |
| 前置条件 | 系统已部署且正常运行，用户已获得访问系统的权限和凭证。 |
| 基本事件流 | <ol style="list-style-type: none"> 1. 打开企业绩效考核系统； 2. 输入用户名和密码； 3. 通过验证进入系统； 4. 系统经过校验获取用户相应权限； 5. 用户通过相应权限进行访问； 6. 用户退出登录。 |
| 其他事件流 | 用户未输入密码及密码错误时系统给出相应的提示信息。 |
| 后置条件 | 用户完成所需的系统操作，数据得到正确的处理和保存，系统提示用户操作完成，用户退出系统并返回主界面。 |

（2）人员管理模块

企业绩效考核系统的人员管理模块是用于管理和维护公司员工个人基本情况。主要功能包括对员工进行新增、修改、按条件查找和删除离职员工等。管理员可以通过该模块管理员工的个人信息，如姓名、邮箱、工号、入职时间和联系方式等。同时，管理员还可以对员工的职务和职级进行管理和调整，以便更好地满足企业的人力资源需求。在企业绩效考核系统中，普通员工没有权限进行人员信息管理，只有管理员和其他被授权的用户才能够进行该功能的操作。用例图如图 3-5 所示，用例功能描述表见表 3-2。

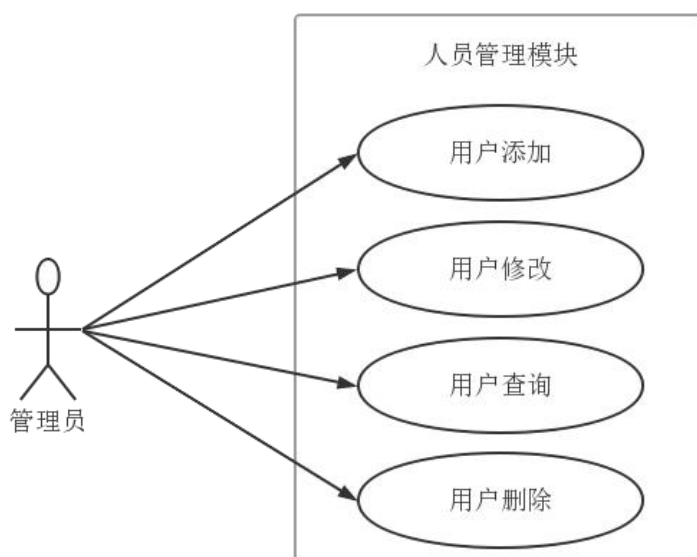


图 3-5 人员管理用例图

表 3-2 人员管理用例功能描述表

| 描述项 | 说明 |
|------|--|
| 用例名称 | 人员管理 |
| 参与者 | 绩效考核系统管理员 |
| 用例描述 | 对企业员工的基本信息进行管理： 1. 新增用户； 2. 修改员工账号名、邮箱等基本信息； 3. 修改员工岗位信息、所属部门信息； 4. 根据用户名等条件查找员工； 5. 对离职员工进行删除。 |

表 3-2 人员管理用例功能描述表（续表）

| 描述项 | 说明 |
|-------|--|
| 前置条件 | 系统在环境下正常运行，且有人管理的实际需求。 |
| 基本事件流 | <ol style="list-style-type: none"> 1. 登录绩效考核系统； 2. 点击界面进入该模块； 3. 根据企业需求对人员基本信息进行信息管理维护； 4. 根据需求对人员所属部门进行修改； 5. 退出系统 |
| 其他事件流 | 对人员进行基本信息操作失败时会提示相应动作失败。 |
| 后置条件 | 对员工的信息管理、维护成功。 |

（3）部门管理模块

该模块是用于修改和维护企业组织架构的功能模块。管理员可以使用该模块进行部门的增加、删除、修改部门、修改部门负责人以及查询部门内的组织结构等操作。部门管理用例图如图 3-6 所示。

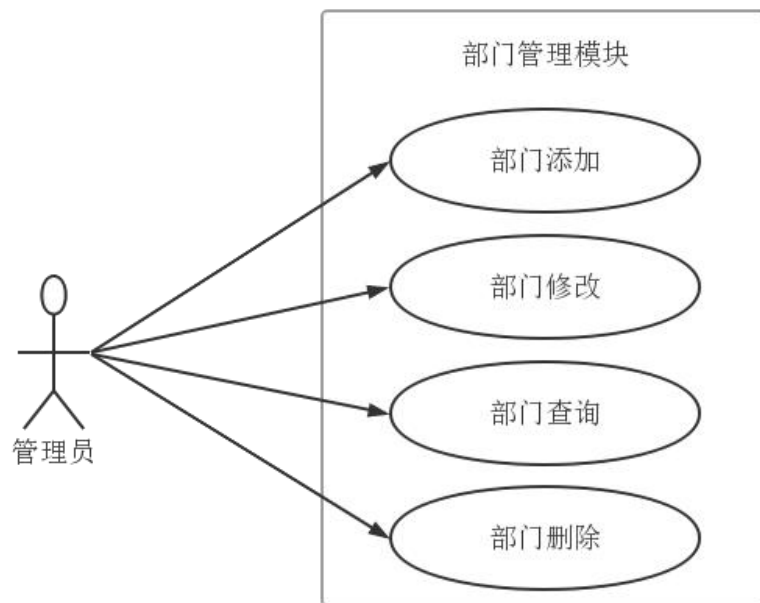


图 3-6 部门管理用例图

（4）岗位管理模块

该模块负责修改和维护公司员工的岗位信息。管理员可以在该模块添加、删除、修改和查询岗位信息。普通员工可以查看自己所在的岗位信息，但是无权进行修改。该模块的设计应充分考虑员工的岗位信息在绩效考核中的重要性，并确保系统能够准确、及时地反映岗位信息的修改和维护。同时，为了保护员工隐私和信息安全，该模块需要实现严格的权限控制和访问限制。岗位管理用例图如图 3-7 所示。

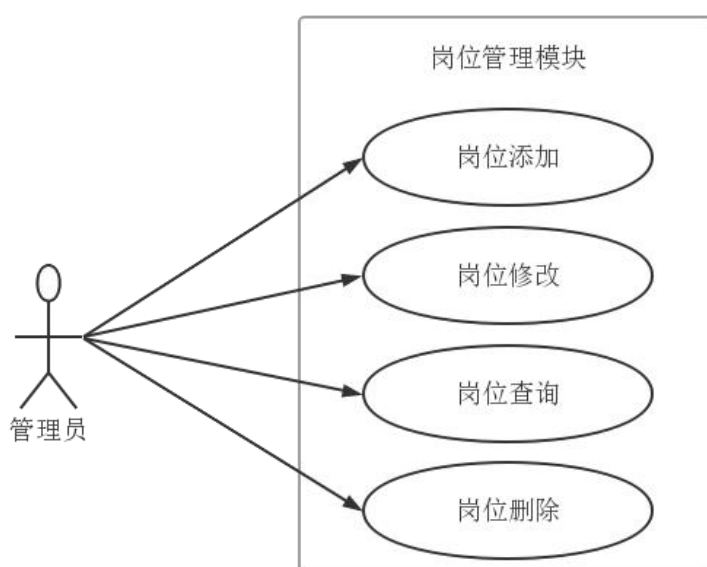


图 3-7 岗位管理用例图

3.3.2 考核评估需求

企业绩效考核系统的主要目的是进行绩效考核，因此该功能模块是整个系统的重中之重。通过获取员工的职级等信息，该模块可以实现对绩效考核全过程的支持。绩效考核模块划分为绩效管理模块和绩效考核管理模块。各模块具体结构如下：

(1) 绩效管理模块

绩效管理模块由高级员工负责管理和维护。主要功能标记参与绩效考核的普通员工绩效考核状态为待审核、已审核或未提交，并获取员工的绩效考核得分趋势等。绩效管理模块案例描述如下表 3-3 所示。

表 3-3 绩效管理用例功能描述表

| 描述项 | 说明 |
|-------|---|
| 用例名称 | 绩效管理 |
| 参与者 | 高级员工、普通员工 |
| 用例描述 | 高级员工负责绩效考核管理，普通员工查看自身考核趋势 |
| 前置条件 | 系统在环境下正常运行，且有绩效管理的实际需求。 |
| 基本事件流 | 1. 高级员工，登录绩效考核系统。 2. 查看部门员工绩效考核是否提交。 3. 对已经提交的考核进行审批。 4. 查询部门内所以员工绩效考核成绩。可指定时间。 5. 退出系统 |
| 其他事件流 | 可以导出考核结果；高级员工可以查看部门员工考核成绩等操作 |
| 后置条件 | 信息维护成功，流程正常进行 |

(2) 绩效考核模块

基于该企业所有员工的考核流程，管理员负责管理此模块，而普通员工和高级员工参与该模块进行绩效考核。普通员工可以进行自评表的新增、删除、修改和查找操作，而高级员工可审批和打分普通员工提交的自评表，同时也具备普通员工的所有功能。如下图 3-8 和 3-9 分别展示了高级员工和普通员工的考核过程。

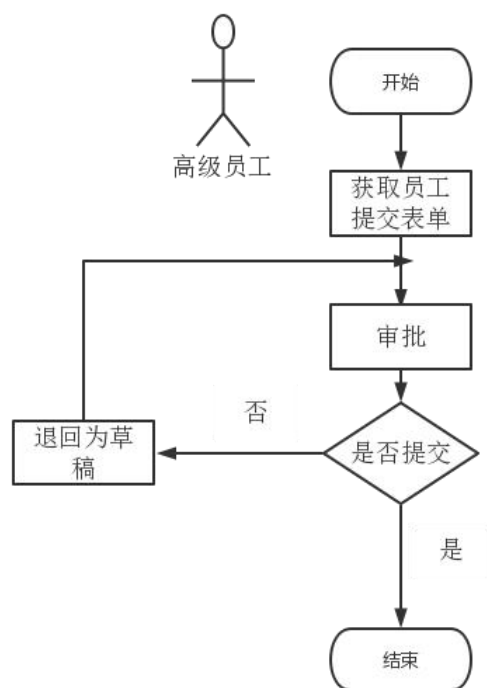


图 3-8 高级员工审批流程图

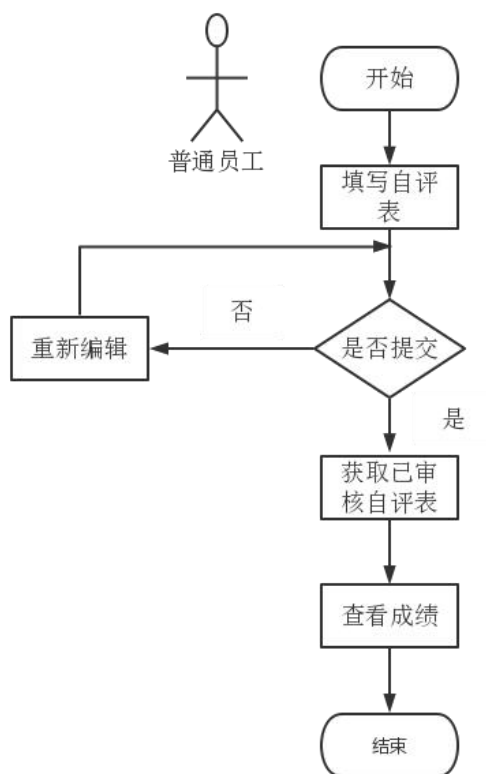


图 3-9 普通员工自评流程图

绩效考核模块描述表见表 3-4。

表 3-4 绩效考核用例功能描述表

| 描述项 | 说明 |
|-------|--|
| 用例名称 | 绩效考核 |
| 参与者 | 高级员工、普通员工 |
| 用例描述 | 高级员工执行绩效审核，而普通员工执行绩效考核自我评定。 |
| 前置条件 | 系统在环境下正常运行，且有绩效考核的实际需求。 |
| 基本事件流 | 1.员工登录绩效考核系统并创建自评表； 2.在自评表中填写评价内容并进行自我评分。 3.可选暂时保存或者直接提交，若直接提交，自评表将流向上级进行评议打分，暂时保存的自评表可以重新编辑。 4.高级员工对自评表进行审批完成 5.普通员工查询绩效考核结果。 |
| 后置条件 | 绩效考核维护成功，绩效考核流程正常进行 |

3.4 非功能性需求分析

(1) 易用性

易用性指的是用户在使用系统时，系统的操作流程简单易懂，用户界面友好、易于操作^[36]。在企业绩效考核系统中，易用性设计应该充分考虑到用户的使用习惯、需求和心理预期，以人性化的理念为设计指导思想。系统前后端的功能设计应该考虑到业务需求，并确保不具备计算机基本知识的员工也能轻松上手，减少新系统的学习难度。最终达到操作简单、使用便捷的目的。

(2) 性能需求

为了确保企业绩效考核系统能够在考核期间稳定高效地运行，需要对系统进行性能优化，主要包括提升系统的存储能力、响应速率和异常处理能力等方面^[37]。在本文所开发的企业绩效考核系统中，必须支持最低 300 人的并发操作，同时需确保页面间的响应时间在 3 秒以内^[38]，以满足用户需求并提高系统使用效率。

(3) 兼容性

企业绩效考核系统是一种典型的 Web 应用程序，目前，主流的浏览器有谷歌、edge、火狐、360 等。考虑到员工的操作环境的多样性，为确保系统的稳定性和可用性，该系统的实现应与运行环境无关^[39]，可以通过一些方法来保证各功能在不同浏览器中正常呈现，从而确保系统的稳定运行。

(4) 安全性

为了确保用户关键的绩效评估内容和个人基本信息的安全性，企业绩效考核系统需要在软件安全方面进行保护。员工需要通过账号和密码登录绩效考核系统。为避免跨站脚本和 SQL 注入等技术攻击，系统需要进行密码加密。此外，系统还需要具备日志打印和数据备份、恢复功能^[40]，以便在出现问题时能够追踪和恢复数据。另外，系统还应保证硬件层面的安全性。

(5) 可靠性

企业绩效考核系统的可靠性是衡量其重要指标之一。企业绩效考核系统不仅是为了提高企业的办公效率和管理效率，同时也是为了使系统比人们更靠谱、更可信^[41]。系统在正式部署上线后，不应该频繁出现异常情况，并且在员工访问高峰期也应该确保系统正常工作。

3.5 本章小结

本章主要对企业绩效考核系统业务需求进行研究。首先介绍了企业绩效考核系统的开发目标、然后分析了系统的企业组织架构和绩效考核流程，最后对绩效考核系统的功能性需求和非功能性需求进行研究。

4 绩效考核系统的设计

本章在前一章需求分析的基础上，对企业绩效考核系统进行了设计。首先，结合 KPI 考核方式设计了该企业各部门的绩效考核指标，然后对系统的总体架构和各功能模块进行了设计和划分，最后进行了数据库表的设计，介绍了主要数据库表。

4.1 绩效考核指标设计

(1) 研发中心绩效考核系统中的绩效考核评价指标主要分为三部分：工作贡献、工作质量、工作态度。如下表 4-1 所示各个绩效考核评价指标提供了衡量标准和评分方法。

表 4-1 研发中心绩效考核方法

| 指标 | 考核内容 | 计分方法 |
|--------------------|---------|--|
| 工作贡献 (总取值 70 分) | 项目成果 | 在考核期内，主导或参与的项目所取得的成果，小分 50 分 |
| | 技术预研成果 | 在考核期内，参与技术方案研究、技术难点攻克、新技术应用，根据成果的重要性和价值给予不同的分值，小分 10 分。 |
| | 其他成果 | 在考核期内，个人在其他方面所做出的工作成果和贡献，包括但不限于参与公司文化建设、领导交办的特殊任务等。根据成果的重要性和价值给予不同的分值，小分 5 分 |
| | 培训及知识分享 | 在考核期内，参与的培训和知识分享活动，小分 5 分 |
| 工作品质 (总取值 15 分) | 项目质量 | 按照要求，保证交付物质量符合标准和规范，包括及时修改和改进不符合要求的交付物，小分 7.5。 |
| | 项目交付 | 按照工作计划，合理安排时间，保证在规定的时间内完成任务，包括及时处理突发事件或问题，小分 7.5。 |
| 工作态度 (总取值 15) | 团队协作 | 积极与其他团队成员沟通合作，及时反馈工作进展情况，主动协助解决问题，促进团队协作和工作效率，小分 7.5 |
| | 工作积极性 | 积极主动完成工作任务，包括主动了解相关业务信息，主动发现和解决问题，小分 7.5。 |

(2) 营销中心绩效考核系统中的绩效考核评价指标主要分为四部分：销售业绩、客户、内部流程、组织学习。如下表 4-2 所示各个绩效考核评价指标提供了衡量标准和评分方法。

表 4-2 营销中心绩效考核方法

| 指标 | 考核内容 | 计分方法 |
|--------------------|---------|---|
| 销售业绩 (总取值 50 分) | 销售额增长率 | 衡量公司销售收入的增长率，包括订单和回款两个维度，分别占 20%的权重。计划期内按照销售额增长率排名，最高的得 20 分，最低的得 0 分，每相差 5%得分上下浮动 1 分。 |
| | 客单价提升率 | 计划期内客户单次购买的平均金额提升率，反映公司产品的市场定位和客户价值，权重 30%。计算方式为（当前客单价-上期客单价）/ 上期客单价*100%。按照提升率排名，最高的得 15 分，最低的得 0 分，每相差 5%得分上下浮动 1 分 |
| | 新客户开发率 | 计划期内新客户占总客户数的比例，反映公司市场拓展的效果，权重 20%。按照新客户开发率排名，最高的得 10 分，最低的得 0 分，每相差 5%得分上下浮动 1 分。 |
| | 销售费用占比 | 计划期内销售费用支出占销售收入比例，反映公司的销售成本管控能力，权重 10%。按照销售费用占比排名，最高的得 5 分，最低的得 0 分，每相差 5%得分上下浮动 1 分。 |
| 客户 (总取值 20) | 客户满意度 | 客户满意度调查结果，反映公司各方面的表现，权重 60%。按照客户满意度排名，最高的得 12 分，最低的得 0 分，每相差 10%得分上下浮动 1 分。 |
| | 客户流失率 | 计划期内失去的客户占总客户数的比例，反映公司客户保持的能力，权重 40%。按照客户流失率排名，最低的得 8 分，最高的得 0 分，每相差 10%得分上下浮动 1 分。 |
| 组织学习 (总取值 20) | 个人学习和发展 | 计划期内，个人参加的培训课程数量和培训效果，以及个人职业规划和发展，反映个人的学习和发展能力，权重 60%。按照个人学习和发展排名，最高的得 12 分，最低的得 0 分，每相差 10%得分上下浮动 1 分。 |
| | 团队合作和协作 | 计划期内，个人和团队之间的协作和合作情况，包括跨部门合作和共享资源，反映个人和团队的合作能力权重 40%。按照团队合作和协作排名，最高的得 8 分，最低的得 0 分，每相差 1 分。 |

表 4-2 营销中心绩效考核方法（续表）

| 指标 | 考核内容 | 计分方法 |
|------------------|---------|--|
| 内部流程 (总取值 10) | 工作效率和质量 | 计划期内，个人和团队的工作效率和质量，包括任务完成率、错误率、回复及时率等指标，权重 100%，小分 10。 |

（3）其他部门绩效考核系统中的绩效考核评价指标主要分为四部分：目标达成度、工作质量、能力提升、其他。如下表 4-3 所示各个绩效考核评价指标提供了衡量标准和评分方法。

表 4-3 其他部门绩效考核方法

| 指标 | 考核内容 | 计分方法 |
|---------------------|--------|---|
| 目标达成度 (总取值 70 分) | 目标完成情况 | 在考核期内，目标达成度是考核员工在一定时间内完成的工作目标的情况。在考核时，应该根据员工的职位和工作职责制定具体的工作目标，按照完成度计分，小分 60 分 |
| | 目标完成质量 | 在考核期内，员工完成目标的质量以及效果，衡量达成的目标对公司业务的贡献，小分 10 分。 |
| 工作质量 (总取值 10 分) | 工作质量 | 在考核期内，衡量员工工作过程中的质量表现，根据员工的工作态度、工作效率、工作准确度、工作规范来计分，小分 10 分。 |
| | 团队协作 | 积极与其他团队成员沟通合作，及时反馈工作进展情况，主动协助解决问题，促进团队协作和工作效率，小分 5 分 |
| 能力提升 (总取值 10 分) | 岗位技能提升 | 通过员工参加技能培训、会议、学习情况计分，小分 5 分。 |
| 其他 (总取值 10 分) | 其他 | 考虑到部门工作的特殊性，可以根据具体情况设置一些其他考核指标，如员工的出勤情况、员工的服务态度，小分 10 分。得分上下浮动 1 分。 |

4.2 总体架构设计

该企业绩效考核系统采用经典的 MVC 三层架构设计思想，遵循 B/S 架构，总体架构如图 4-1 所示。该系统使用 Spring Boot+Vue 框架来构建项目^[42]。系统架构分为视图层（View）、控制层（Controller）、业务逻辑层（Service）和数据持久层（Dao）。数据库服务器主要使用 MySQL 和 Redis 两种数据库。Redis 是一种非关系型数据库，可以有效地解决 MySQL 在大量访问时性能下降的问题，

从而提高了数据的读写速度。视图层使用 Vue 框架和 Element UI 界面渲染工具构建前端页面，前端页面使用 HTTP 协议向控制层发送请求。控制层和业务逻辑层使用 Spring Boot 框架进行构建。控制层接收前端请求，对数据进行解析和验证，然后将数据发送到服务层进行业务逻辑处理。最后，数据持久层会访问数据库服务器进行数据库操作，处理完毕后将结果返回给控制层，控制层将结果发送到前端进行页面处理。

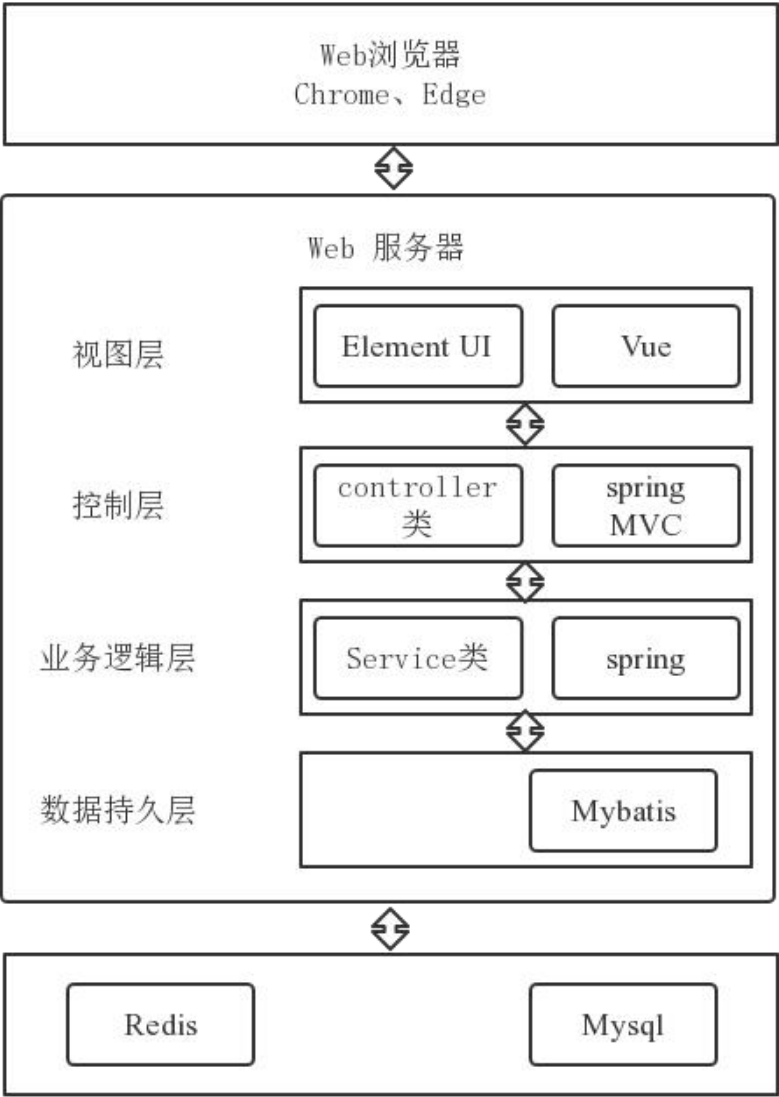


图 4-1 系统架构图

4.3 系统功能模块设计

通过对企业绩效考核系统的需求分析，确定企业绩效考核系统功能结构应包含的系统管理和考核评估两大功能模块。系统管理模块包括系统登录、人员管理、部门管理、岗位管理四个子模块。考核评估模块包括绩效管理和绩效考核两个子模块。这些模块分别满足了企业绩效考核系统的功能性需求。如图 4-2 所示，企业绩效考核系统总体功能结构如下：

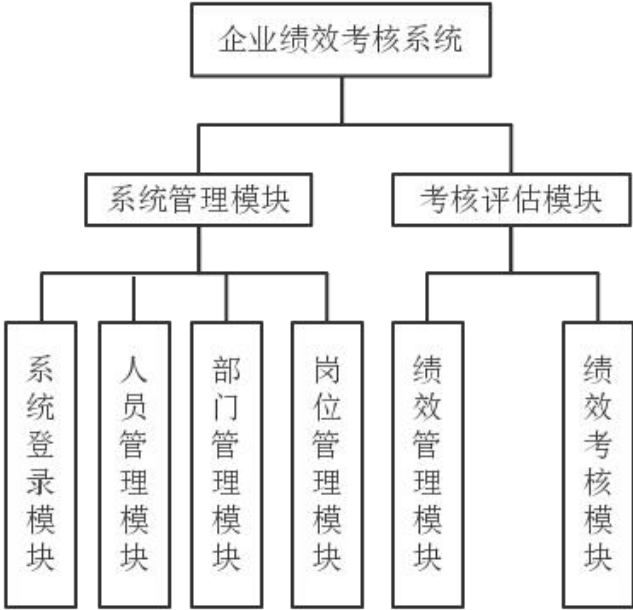


图 4-2 系统功能结构设计图

（1）系统登录功能模块：对应了系统的登录校验管理功能需求，本模块主要是校验登录用户的合法性，保证系统的安全性。

（2）人员管理功能模块：该模块主要是为系统的管理员提供对公司员工基本信息的增删改功能，同时还提供了员工所属部门和权限管理等功能。以满足系统的人员信息管理需求。

（3）部门管理功能模块：该模块主要是为系统的管理员实现了公司部门信息管理功能。以满足系统的部门信息管理功能需求。

（4）岗位管理功能模块：该模块主要是为系统的管理员实现了公司岗位信息管理功能。以满足系统的岗位信息管理功能需求。

(5) 绩效管理功能模块：该模块主要是为系统的普通员工和高级员工提供了考核成绩查看、考核链路完成状态查看和修改等操作功能。以满足系统的绩效管理功能需求。

(6) 绩效考核功能模块：该模块主要是为系统的普通员工和高级员工提供了填写自评表、撤回自评表、执行绩效审核等操作功能。以满足系统的绩效考核管理功能需求。

4.4 系统数据库设计

使用数据库来存储和处理系统中的所有数据，有助于减少程序开发人员的工作量，同时也可以实现数据的共享和提高系统的可扩展性和灵活性。该系统选择 MySQL 数据库进行数据存储、采用 Redis 进行数据缓存。数据库设计应根据对企业绩效考核的需求分析，对系统的概念结构设计，再优化设计逻辑结构，最后进行数据库的物理结构，从而提高系统数据查询效率和稳定性。

4.4.1 概念结构设计

概念模型是基于对企业绩效考核系统的关键流程分析，通过实体-属性 E-R(Entity-Relationship)图表示，分析了企业绩效考核系统中主要实体、属性和联系^[43]。如下图图 4-3 所示。系统用户主要包括管理员，普通员工，高级员工，属于人员实体，管理员可以管理部门、人员、岗位、职位等实体，人员和职位，岗位，考核内容存在一对多的关系，考核内容和考核结果存在一对一的关系。

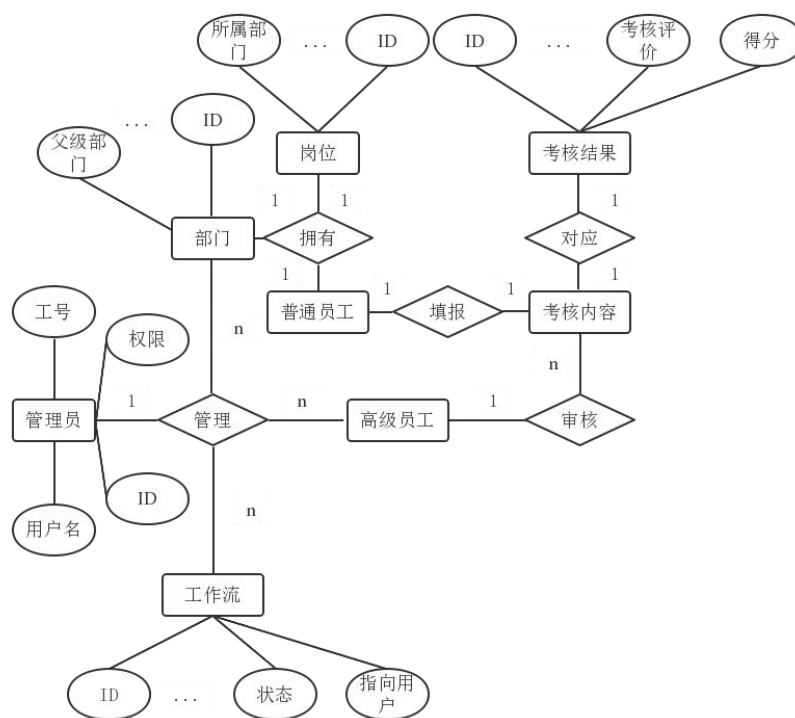


图 4-3 企业绩效考核系统 E-R 图

4.4.2 逻辑结构设计

基于上文中得到的关系模式，关系数据库的构造应满足范式的规则。目前关系数据库有六种范式：第一范式（1NF）、第二范式（2NF）、第三范式（3NF）、巴斯范式（BCNF）、第四范式（4NF）和第五范式（5NF）。按照以上关系模型优化的规则，将本系统中的关系模型进行优化得到数据库逻辑设计内容如下：

人员关系模式（员工编号，用户名，邮箱，工号，密码，姓名，部门编号，职位编号，职称，入职时间，权限，执行链编号）；

部门关系模式（部门编号，上级部门，部门名称，部门树，部门说明，成立时间）；

岗位关系模式（岗位编号，岗位名，岗位说明，创建时间）；

考核内容关系模式（执行链编号，考核模块，考核指标，目标，权重，加分项，减分项）；

考核结果关系模式（考核编号，考核评价，员工自评得分，上级得分，部门得分，考核得分）；

workflow关系模式（ workflow编号，创建用户，链路编号，指向用户，状态，完成时间）；

管理关系模式（人员编号，部门编号，岗位编号， workflow编号）；

拥有关系模式（人员编号，部门编号，岗位编号）；

填报关系模式（人员编号，考核编号）；

审核关系模式（人员编号，考核编号）；

对应关系模式（执行链编号，考核编号）。

4.4.3 物理结构设计

基于上文中的系统数据 E-R 模型和关系模型，设计了绩效考核系统各功能模块对应的数据库表，核心表的物理结构^[44]。包括字段名、数据类型、字段描述、主键、外键、是否为空和是否能重复。

(1) 用户信息表（jmoa_user）

如下表 4-5 所示为用户信息表，该表主要包含用户 id、用户名、部门 id、密码、职级、人员管理权限、绩效执行链 id 等。

表 4-4 用户信息表

| 字段名 | 数据类型 | 备注 | 描述 |
|---------------|---------|---------|-------|
| id | int | 主键，自增 | 用户 id |
| user_name | varchar | 不为空，不重复 | 用户名 |
| mail | varchar | 不为空，不重复 | 邮箱 |
| work_num | varchar | 不为空，不重复 | 工号 |
| password | varchar | 不为空，加密 | 密码 |
| real_name | varchar | 不为空，不重复 | 姓名 |
| dept_id | int | 外键，不为空 | 部门 id |
| user_level_id | int | 外键，不为空 | 职级 id |

表 4-4 用户信息表（续表）

| 字段名 | 数据类型 | 备注 | 描述 |
|-------------|----------|--------|----------|
| job_name | varchar | | 职称 |
| entry_time | datetime | 不为空 | 入职时间 |
| permissions | int | 不为空 | 人员管理权限 |
| is_resign | int | 不为空 | 是否离职 |
| is_ignore | int | 不为空 | 是否需要填写绩效 |
| chain_id | int | 外键，不为空 | 执行链 id |
| update_time | datetime | 不为空 | 更新时间 |
| create_time | datetime | 不为空 | 创建时间 |
| bz | varchar | | 备注 |

(2) 部门信息表（jmoa_dept）

如下表 4-6 所示为部门信息表。该表主要包含部门名称、id、父级部门、部门树、创建时间等。

表 4-5 部门信息表

| 字段名 | 数据类型 | 备注 | 描述 |
|-------------|----------|-------|---------|
| id | int | 主键，自增 | 部门 id |
| pid | int | 不为空 | 父级部门 id |
| dept_name | varchar | 不为空 | 部门名称 |
| dept_tree | varchar | 不为空 | 部门树 |
| dept_desc | varchar | 不为空 | 部门说明 |
| update_time | datetime | 不为空 | 更新时间 |
| create_time | datetime | 不为空 | 创建时间 |
| bz | varchar | | 备注 |

(3) 绩效链路表（jmoa_chain）

如下表 4-7 所示为绩效链路表，主要包含绩效链路 id、链路名称、链路说明、链路流程等。

表 4-6 绩效链路表

| 字段名 | 数据类型 | 备注 | 描述 |
|-------------|----------|-------|-------|
| id | int | 主键，自增 | 链路 id |
| chain_name | varchar | 不为空 | 链路名称 |
| chain_desc | varchar | 不为空 | 链路说明 |
| chain_flow | varchar | 不为空 | 链路流程 |
| update_time | datetime | 不为空 | 更新时间 |
| create_time | datetime | 不为空 | 创建时间 |
| bz | varchar | | 备注 |

(4) 绩效考核表 (jmoa_performance)

如下表 4-8 所示为绩效考核表：主要包含绩效考核 id、用户 id、月份、自评得分、上级得分、加减分、权重、考核得分、考核等级等字段。该表主要展示个人考核结果。

表 4-7 绩效考核表

| 字段名 | 数据类型 | 备注 | 描述 |
|-------------------|---------|--------|-------|
| id | int | 主键，自增 | 绩效 id |
| user_id | int | 外键，不为空 | 用户 id |
| month | varchar | 不为空 | 月份 |
| score_self | double | 不为空 | 自评得分 |
| score_super | double | 不为空 | 上级得分 |
| score_branch | double | 不为空 | 部门得分 |
| score_final | double | 不为空 | 考核得分 |
| level | int | 不为空 | 考核等级 |
| state | int | 不为空 | 审核状态 |
| pluses_content | text | | 加分项内容 |
| pluses_num | int | | 加分项数值 |
| reduction_content | text | | 减分项内容 |
| reduction_num | int | | 减分项数值 |

| | | | |
|-------------|----------|--------|--------|
| evaluation | text | 不为空 | 评价 |
| update_time | datetime | 不为空 | 更新时间 |
| create_time | datetime | 不为空 | 创建时间 |
| bz | varchar | | 备注 |
| weight | tinyint | 不为空 | 权重 |
| super_name | varchar | 不为空 | 直接上级 |
| chain_id | int | 外键，不为空 | 执行链 id |

(5) 绩效信息表 (jmoa_performance_item)

如下表 4-8 所示为绩效考核信息表，包含被考核者各考核项目信息，主要有考核模块、实际完成情况、得分等字段。

表 4-8 绩效信息表

| 字段名 | 数据类型 | 备注 | 描述 |
|--------------|----------|--------|---------|
| id | Int | 主键，自增 | id |
| per_id | int | 外键，不为空 | 绩效 id |
| line | int | 不为空 | 行 |
| model | text | 不为空 | 考核模块 |
| indicator | text | 不为空 | 考核指标 |
| target | text | 不为空 | 目标 |
| weight | double | 不为空 | 权重 |
| complete | text | 不为空 | 实际完成情况 |
| score_self | double | 不为空 | 员工自评分 |
| score_super | double | 不为空 | 直接上级评分 |
| score_branch | double | 不为空 | 部门负责人评分 |
| update_time | datetime | 不为空 | 更新时间 |
| create_time | datetime | 不为空 | 创建时间 |
| bz | varchar | | 备注 |

(6) 工作流表 (jmoa_work_flow)

如下表 4-9 所示为工作流表。

表 4-9 workflow信息表

| 字段名 | 数据类型 | 备注 | 描述 |
|-------------|----------|--------|-------------|
| flow_id | varchar | 主键，自增 | workflow id |
| submit_user | int | 不为空 | 创建用户 |
| chain_id | int | 外键，不为空 | 链路 id |
| chain_next | int | 不为空 | 指向用户 |
| state | int | 不为空 | 状态 |
| per_id | int | 外键，不为空 | 绩效 id |
| update_time | datetime | 不为空 | 更新时间 |
| create_time | datetime | 不为空 | 创建时间 |
| bz | varchar | | 备注 |

4.5 本章小结

本章首先设计了绩效考核系统的总体结构，然后对各个子模块进行详细设计，并采用用例图和用例描述表两种方式描述对系统各个模块，最后设计实现了企业绩效考核系统的数据库，利用 E-R 图设计概念模型。然后通过逻辑结构设计和物理结构设计，展示了核心的数据字典和物理结构。

5 系统的实现与测试

本章基于前两章的企业绩效考核系统需求分析以及绩效考核系统的详细设计，完成企业绩效考核系统相关的功能模块开发，并完成系统的功能测试和非功能测试。

5.1 系统环境部署

在选择系统开发环境时，需要根据不同的技术路线进行考虑，以便能够选择最适合的环境，确保开发过程的高效性和可靠性。系统的软件开发环境如下表 5-1 所示：

表 5-1 系统软件开发环境

| 分类 | 名称 | 版本 |
|----------|---|--------|
| 操作系统 | Windows10 | |
| 开发语言 | Java | JDK1.8 |
| 开发工具 | IntelliJ IDEA 2020、 Visual Studio Code | |
| 数据库缓存服务器 | Redis | 3.6 |
| 数据库平台 | MySQL | 5.5.40 |
| 浏览器 | Internet Explorer 、 Chrome、Firefox 等 | |

本文基于前后端分离架构设计开发企业绩效考核系统。在系统软件环境方面，选用 IntelliJ IDEA 2020 作为后端开发工具，并使用主流的开发语言 Java 作为系统开发语言，使用 JDK1.8，后端采用 Spring boot 框架，同时整合 MySQL 和 Redis 等技术。前端开发工具采用 Visual Studio Code，并使用 Axios 技术与后端通信。受成本限制，选取了企业电脑作为开发机和系统部署服务器，具体被配置为处理器：Intel（R） Core（TM） I5-10400，内存：16GB，硬盘：256GB，网络：100M 局域网。

5.2 系统模块实现

5.2.1 系统登录模块

绩效考核系统登录模块是最基础、最重要的模块之一。在实现这一模块时，需要保证用户可以安全地登录系统，同时需要对输入的用户名和密码进行校验，以确保系统的安全性。如下图 5-1 所示：



图 5-1 用户登录界面

如表 5-2 所示为系统登录模块主要类：

表 5-2 系统登录模块主要类表

| | |
|-------|------------------|
| 控制层类 | LoginController |
| 业务逻辑类 | LoginService |
| | LoginServiceImpl |
| 持久对象类 | redisSave |
| | User |

当员工在系统登录界面输入账号和账号密码后，前端会调用登录接口传递给后端。考虑到员工登录时存在高并发的情况，后端将通过 redisSave 方法将接受的数据存储在 Redis 中，然后 LoginController 中的 login 方法会对这些信息进行

处理，并调用 LoginServiceImpl 中的具体实现来完成实际的业务逻辑。最终，后端会返回一个 token 作为数据交换的凭证。

5.2.2 人员管理模块

该模块实现员工信息维护的功能，人员信息模块提供添加、删除、更新和查询人员信息的功能。管理员可以通过该模块管理人员情况。具体来说，管理员可以添加新的员工，删除现有的员工，更新员工，以及查询员工。此功能在系统的人力资源管理方面具有重要意义。人员管理的详情界面如图 5-2 所示：

湖南计算机公司(6)

财务部(1)

综合部(0)

研发中心(2)

营销中心(1)

供应链中心(1)

姓名

搜索

重置

新增员工

| 姓名 | 公司邮箱 | 部门 | 岗位 | 级别 | 操作 |
|------|------------------|---------|-----------|------|-------------------------------------|
| 胡八 | lb@jmdas.com | 供应链中心 | 采购经理 | 经理 | ✕ 📄 |
| 李四 | 12@163.com | 测试部 | 测试副总监 | 经理 | ✕ 📄 |
| 张一 | 1s2a@qq.com | 财务部 | 生成 | 员工 | ✕ 📄 |
| joiy | 22@189.com | 营销中心 | 副总经理 | 副总经理 | ✕ 📄 |
| 胡靖宇 | 11@qq.com | 软件部 | java开发工程师 | 员工 | ✕ 📄 |
| 王五 | 20201212@163.com | 湖南计算机公司 | CEO | CEO | ✕ 📄 |

图 5-2 人员管理详情界面

根据图 5-3 和图 5-4 所示，管理员在进行员工信息的添加和修改时，需要填写姓名、部门、账号、岗位、职级、邮箱等必填字段。此外，管理员还可以选择该员工是否有管理机构，并且可以添加绩效汇报链路。在实现过程中，当管理员获取到员工的相关信息，然后通过后端提供的接口将这些信息发送到后端进行处理，最后将处理结果返回给前端显示或者将处理结果保存到数据库中。

编辑员工

×

姓名

胡八

部门

供应链中心

账号

lb

工号

4

岗位

采购经理

职级

经理

邮箱

lb@jmdas.com

入职时间

2021-09-03

管理机构

4个

+

提交绩效

☒

绩效汇报

王五

+

人员管理权限

☐

取消

确定

图 5-3 修改员工信息详情界面

新增员工

×

姓名

部门

账号

工号

岗位

职级

员工

邮箱

入职时间

2022-10-12

管理机构

0个

+

提交绩效

☒

绩效汇报

+

人员管理权限

☐

取消

确定

图 5-4 添加员工信息详情界面

人员管理模块由于包含用户部门, 和职级, 通过连表查询的方式, 其主要类如下:

表 5-3 人员管理模块主要类表

| | |
|-------|----------------------|
| 控制层类 | AccountController |
| 业务逻辑类 | UserService |
| | UserDeptService |
| | UserLevelService |
| | UserServiceImpl |
| | UserDeptServiceImpl |
| 持久对象类 | UserLevelServiceImpl |
| | User |
| | UserDept |
| 数据持久类 | UserLevel |
| | UserMapper.xml |
| | UserDeptMapper.xml |
| | UserLevelMapper.xml |

当管理员在前端页面新增员工时, 前端会填写员工的基本信息, 通过请求 /oa/user 接口将数据发送到后端的 UesrController 进行处理。UesrController 调用 UserServiceImpl 中的 savaUser 方法, 将员工数据存储到数据库中。新增员工时还通过 UserLevel 和 UserDept 将用户与其自身职级、所在部门进行关联, 用户信息也将通过 UserMapper 写入数据库。

5.2.3 部门管理模块

该模块主要功能是对部门信息修改。如下表 5-4 为该模块主要类:

表 5-4 部门管理模块主要类表

| | |
|-------|----------------------------|
| 控制层类 | DepartmentController.java |
| 业务逻辑类 | DepartmentService.java |
| | DepartmentServiceImpl.java |
| 持久对象类 | Department.java |
| 数据持久类 | DepartmentMapper.xml |

在新增部门时, 前端将用户填写好的数据, 通过/oa/dept 接口交给后端的 DepartmentController, 接收数据后 DepartmentServiceImpl 的 savaOrg 方法实现这一业务逻辑。部门信息也将通过 DepartmentMapper 写入数据库。

5.2.4 岗位管理模块

该模块提供岗位信息的维护功能，管理员可以对岗位信息进行增删改查操作。
如下表 5-5 为该模块包含的类：

表 5-5 岗位管理模块主要类表

| | |
|-------|----------------|
| 控制层类 | JobController |
| 业务逻辑类 | JobService |
| | JobServiceImpl |
| 持久对象类 | Job |
| 数据持久类 | JobMapper.xml |

在新增用户时，前端将用户填写好的数据，通过/oa/job 接口交给后端的 JobController，接收数据后 jobServiceImpl 的 savaJob 方法实现这一业务逻辑。岗位信息也将通过 JobMapper 写入数据库。

5.2.5 绩效管理模块

绩效管理模块由管理员和高级员工进行管理和维护。该模块主要功能有对普通员工的绩效考核状态进行标记待审核、已审核、未提交，获取员工的绩效考核得分趋势等。其界面图如下图所示：

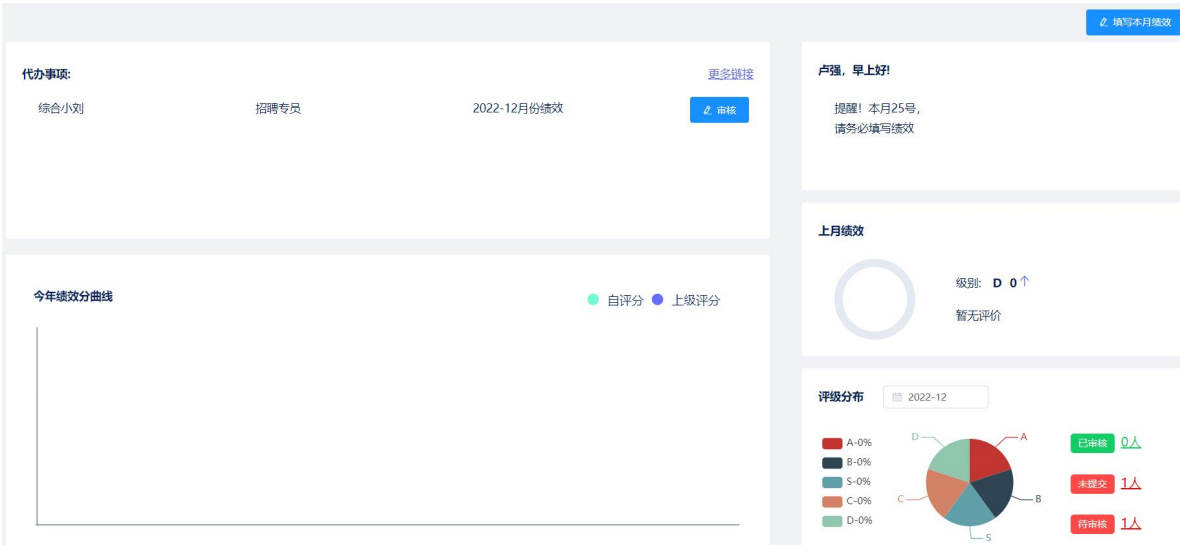


图 5-5 绩效管理详情界面



图 5-6 查看未提交考核界面



图 5-7 考核结果界面

该模块主要包含的类如下表 5-6 所示：

表 5-6 绩效管理模块主要类表

| | |
|-------|-----------------------------|
| 控制层类 | PerformanceController.java |
| 业务逻辑类 | PerformanceService.java |
| | WorkFlowService.java |
| | PerformanceServiceImpl.java |
| 持久对象类 | WorkFlowServiceImpl.java |
| | Performance.java |
| | WorkFlow.java |
| 数据持久类 | PerformanceMapper.xml |
| | WorkFlowMapper.xml |

在绩效管理时，前端将用户填写好的数据，通过/oa/performance 接口交给后端的 PerformanceController，接收数据后 PerformanceServiceImpl 的 unhandled、gragh 等方法实现这一业务逻辑。

5.2.6 绩效考核模块

在绩效考核模块中，高级员工和普通员工都有操作权限，只是高级员工需要审核普通员工的自评表，提交绩效考核自评表如下图 5-8 所示：

| 月度绩效考核表 | | | | | | | | | | | |
|---------|--|----|-----|------|-----------|-------------------------|------------|------|-------|--------|---------|
| 姓名 | 李四 | 部门 | 测试部 | 岗位 | 测试副总监 | 入职时间 | 2022-12-03 | 直接上级 | 王五 | 考核月份 | 2023-01 |
| 考核模块/项目 | 考核指标(关键任务) | | | 目标 | 权重(单位: %) | 实际完成情况(可从时间/质量/成本等角度描述) | | | 员工自评分 | 直接上级评分 | 部门负责人评分 |
| 项目成果 | 主导或参与的项目所取得的成果 | | | 50分 | | | | | | | |
| 技术钻研成果 | 参与技术方案研究、技术难点攻克、新技术应用 | | | 10分 | | | | | | | |
| 其他成果 | 个人在其他方面所做出的工作成果和贡献 | | | 5分 | | | | | | | |
| 培训及分享 | 参与的培训和知识分享活动 | | | 5分 | | | | | | | |
| 项目质量 | 保证交付物质量符合标准和规范，包括及时修改和改进不符合要求的交付物 | | | 7.5分 | | | | | | | |
| 项目交付 | 按照工作计划，合理安排时间，保证在规定的时间内完成任务，包括及时处理突发事件或问题 | | | 7.5分 | | | | | | | |
| 团队协作 | 积极与其他团队/成员沟通合作，及时汇报工作进展情况，主动协助解决问题，促进团队协作和工作效率 | | | 7.5分 | | | | | | | |
| 工作积极性 | 积极主动完成工作任务，包括主动了解相关业务信息，主动发现和解决问题 | | | 7.5分 | | | | | | | |

图 5-8 绩效考核自评表界面

普通员工在提交自评表，填写必填字段，提交表单参与审核，主要类如下表 5-7 所示：

表 5-7 绩效考核模块主要类表

| | |
|-------|-----------------------------|
| 控制层类 | PerformanceController.java |
| 业务逻辑类 | PerformanceService.java |
| | WorkflowService.java |
| | PerformanceServiceImpl.java |
| 持久对象类 | WorkflowServiceImpl.java |
| | Performance.java |
| | Workflow.java |
| 数据持久类 | PerformanceMapper.xml |
| | WorkflowMapper.xml |

当新增绩效时，前端将用户填写的数据通过/oa/performance 接口传递给后端的 PerformanceController。后端 PerformanceServiceImpl 的 savePerformance 方法接收数据并实现相应的业务逻辑，绩效考核信息将通过 PerformanceMapper 存储到数据库中。

5.3 系统测试

系统的软件测试是保证软件质量的重要环节之一。在软件开发过程中，通过使用各种测试方法和技术，可以发现潜在的错误、缺陷和问题，以确保软件的功能正常、可靠且符合用户需求。

5.3.1 系统测试方法

系统测试主要包括单元测试、集成测试、系统测试和验收测试。单元测试是对软件系统中最小的可测试单元进行测试，通常是对代码中的函数、方法或模块进行独立测试。目的是验证每个单元的功能是否按照设计要求正确执行。集成测试是将多个已经通过单元测试的模块组合在一起进行测试。测试人员验证各个模块之间的接口和交互是否正确，以确保集成后的软件系统能够正常工作。系统测试是对完整的软件系统进行全面测试，以验证系统在各种使用情况下的功能、性能、稳定性和可靠性。测试人员模拟真实的使用环境和用户场景，测试整个系统的功能和交互。验收测试是由最终用户、客户或授权代表进行的测试，目的是确认软件是否满足用户的需求和预期。这种测试通常在软件开发的最后阶段进行，以确保软件能够按照合同和规格要求交付给客户。

单元测试通常由开发人员在编写代码时完成，采用白盒测试方法，以确保代码的正确性和功能的准确性。开发人员对代码的内部结构和逻辑有较深的了解，可以使用白盒测试方法进行测试，覆盖各个代码路径和逻辑分支，以发现潜在的错误和问题。系统测试和验收测试则主要采用黑盒测试方法，以验证整个软件系统在用户角度下的功能、性能、稳定性和可靠性。测试人员在这些测试中不需要了解系统的内部实现细节，而是关注系统的输入和输出，以及系统对输入的处理和响应。通过对各种输入情况进行测试，包括正常值、异常值和边界值，以及不同输入组合，可以验证系统是否能够正确处理输入并给出正确的输出和提示信息。

5.3.2 功能性测试

员工的信息管理模块是系统的基础模块，用户登录后，具有人员权限的用户可以通过人员管理模块下的操作按钮。从而得到最终结果表 5-8 所示为员工的测试用例和结果。

表 5-8 员工信息测试用例和结果

| 编号 | 测试项目 | 操作步骤 | 预期结果 | 实际结果 |
|----|----------|-------------|---------------------|------|
| 1 | 修改员工 | 修改用户的基本信息 | 人员管理界面该用户信息变更为最新 | 成功 |
| 2 | 添加员工 | 按照页面步骤添加信息 | 新添加的用户能够正常显示在人员管理界面 | 成功 |
| 3 | 设置登录名和密码 | 输入正确的登录名和密码 | 用户将正常登录到绩效考核系统 | 成功 |

高级员工通过登录入口进入系统，在考核评估模块下，通过前端提供的操作按钮，对部门内普通员工绩效考核信息管理维护。表 5-9 所示为绩效管理维护的测试用例和结果。

表 5-9 绩效管理维护测试用例和结果

| 编号 | 测试项目 | 操作步骤 | 预期结果 | 实际结果 |
|----|----------|------------|--------------------|------|
| 1 | 查看未审核 | 按照页面步骤添加信息 | 未审核考核记录界面正常显示 | 成功 |
| 2 | 查看部门考核结果 | 按照页面步骤添加信息 | 部门内所有员工考核结果在页面正常显示 | 成功 |
| 3 | 查看未提交 | 按照页面步骤添加信息 | 未提交考核记录界面正常显示 | 成功 |

员工通过登录入口进入系统，进行绩效考核，通过前端提供的操作按钮，填写自评表，可暂存重新编辑，然后提交给高级员工审核。高级员工负责审核普通员工自评表，从而得到最终结果表 5-10 所示为绩效考核维护的测试用例和结果。

表 5-10 绩效考核测试用例和结果

| 编号 | 测试项目 | 操作步骤 | 预期结果 | 实际结果 |
|----|--------|------------|-----------------|------|
| 1 | 审批考核表 | 按照页面步骤添加信息 | 考核审批完毕，考核得分正常显示 | 成功 |
| 2 | 编辑绩效考核 | 按照页面步骤添加信息 | 填写自评表正常上传 | 成功 |
| 3 | 导出考核结果 | 根据页面步骤添加信息 | excel 结果正常完整导出 | 成功 |

5.3.3 非功能性测试

(1) 系统的性能测试

系统的使用人数和数据量、任务量并不是很大，该系统同时采用 Redis 缓存机制，可以支持较高的并发量。用户访问每个模块的页面跳转时间都不超过 3 秒^[45]。

(2) 系统易用性测试

企业绩效考核系统采用 MVVM 前端设计模式，满足了用户常用的流程和逻辑。同时，该界面简单、美观、遵循用户习惯，满足易用性要求。

(3) 系统兼容性测试

由于目前市面上浏览器种类繁多，且浏览器版本类型也比较多，由于该企业绩效考核系统是基于 B/S 架构的 WEB 系统，所以兼容性问题一般指的是浏览器是否兼容^[46]，因此本文对主流的浏览器 Chrome、火狐、MicroSoft Edge、360 浏览器进行各模块测试，页面均正常显示、功能均可正常使用。

5.4 本章小结

本章首先介绍系统部署的软硬件开发环境，着重阐述了第四章中设计的企业绩效考核系统重要模块的界面和功能实现。然后通过测试用例等方法，对企业绩效考核系统的功能性进行测试，并从系统性能、兼容性、易用性三方面进行非功能测试，企业绩效考核系统的测试结果满足预期目标。

6 总结与展望

6.1 全文总结

本文在对课题背景进行分析后,详细介绍了企业绩效考核系统的设计和实现,以及对其研究背景、意义以及国内外研究现状进行全面阐述。通过需求分析、概要设计和数据库设计,本文具体分析了企业现有绩效考核模式存在的问题,进而实现了该系统。本文具体研究工作如下:

(1) 采用目前主流的前后端分离的开发方式,将后端与前端分离,让两者各司其职,可以更好地实现系统的可扩展性和可维护性。在本文的绩效考核系统的设计与实现中,采用了 SpringBoot 框架作为后端开发的主要框架,它可以快速构建基于 Spring 框架的 Web 应用程序,并且可以很好地与其他框架和技术集成。同时,采用 Vue+Element-UI 来构建 SPA 即单页面应用作为前端开发的主要技术,这可以让用户获得更好的用户体验。Vue 是一个轻量级的 JavaScript 框架,具有简单易用的 API 和出色的性能,而 Element-UI 是一个基于 Vue.js 的框架,提供了大量的 UI 组件和操作,可以很好地提高开发效率和用户体验。综合使用这些技术,可以更好地实现系统的可扩展性和可维护性,为系统的长期发展打下了坚实的基础。

(2) 针对绩效考核系统进行业务需求和非功能性需求,并对绩效考核指标体系进行了评估。基于系统的需求分析结果,明确了系统的总体架构、各功能模块、数据库结构。采用 Ajax 技术来实现页面数据的异步交互。后端方面采用 MVC 三层架构,而前则采用 MVVM 的开发设计模式。

(3) 通过系统的分析设计功能模块包括考核评估和系统管理两个子模块。然后对数据库进行建模和表设计,并详细分析了表之间的关系和重要字段。

(4) 针对企业绩效考核系统的设计,详细介绍了企业绩效考核系统实施过程。对系统性测试。测试结果显示,系统整体上符合预期要求,虽然存在一些问题,但这些问题不会影响系统的功能。后期会对这些问题进行优化。

6.2 工作展望

尽管绩效考核系统已经在公司稳定运行，并成功的调动了员工工作积极性，同时也为员工的晋升提供了数据基础，但仍存在以下不足之处：

（1）目前绩效考核系统仅在公司内网部署，暂时未能在互联网端开放访问端口，这限制了在外地或家里使用绩效考核系统的员工。因此，在下一阶段的规划与设计中，需要增加安全防护措施并开放外网访问权限。

（2）绩效考核系统基于 Web 模式开发，主要运行在 PC 端，但当前移动端用户比例大幅提升。为了满足员工使用习惯，下一阶段的升级将增加移动端支持，使员工可以在手机等移动设备上使用绩效考核系统。

（3）随着企业的不断发展以及企业绩效考核的完善，需要进一步的改进和升级企业绩效考核系统，以满足不断变化的业务需求和技术发展趋势，从而保持系统的竞争力和可持续发展。

参考文献

- [1] J. Qu, F. Dang, Y. Xu , F. Fan.Design and Implementation of Power Enterprise Staff Performance Appraisal System Based on Flex and J2EE[C]// International Conference on Computational & Information Sciences.IEEE, 2012.
- [2] 杨雪梅.企业绩效考核现状研究[J].现代交际,2011(09):17-18.
- [3] 郑楚华, 李炜兴. 国外企业绩效考核研究综述[J]. 中小企业管理与科技(上旬刊), 2014(6): 16-16.
- [4] 唐安双, 高赛男. 全员绩效考核体系在实践中的思考[J]. 现代商贸工业, 2014, 26(14): 76-77.
- [5] 言儒斌. 基于 Oracle ERP 的银行薪酬管理系统的设计与实施[D]. 济南: 山东大学, 2015.
- [6] Ghosh S M, Sharma H R, Mohabay V. Study of Impact Analysis of Software RequirementChange in SAP ERP[J]. International Journal of Advanced Science & Technology, 2011, 32.
- [7] 何洋. 国内外绩效考核现状研究[J]. 现代商业, 2014(35):139-140.
- [8] 蒋丽春. 浅析我国企业绩效管理存在的问题及对策[J]. 中国集体经济, 2010(18):57-58.
- [9] 邱红. 浅析金蝶 ERP 系统在 W 企业的运用 [J]. 财会学习, 2017(17):210-210.
- [10]刘莹. 用友 ERP 财务管理软件使用探析[J]. 商业会计, 2016(20):104-105.
- [11]李顺章. 基于平衡计分的绩效考核系统的设计与实现[D]. 西安: 西安电子科技大学, 2017.
- [12]陈大祥. 人力资源管理[M].广州.中山大学出版社,2001: 41~43.
- [13]张明. 浅析员工 360 度考核在企业中的应用[J]. 人力资源管理, 2014(10):153-153.
- [14]Issar G, Navon L R. Key Performances Indicators (KPI)[J]. 2016.
- [15]李运霞. 平衡计分卡在企业绩效管理中的应用[J]. 北极光, 2016(8).
- [16]Suryotkend Application for Public Complaint Systems Using MicroserviceSpringBoot[J]. Procedia Computer Science, 2017, 124:736-743.

- [17]易剑波.基于 MVVM 模式的 WEB 前端框架的研究[J].信息与电脑（理论版）. 2016（19）： 76-77+84 .
- [18]Hall, Mclean G . Pro Wpf and Silverlight MVVM: Effective Application Development with Model-View-Viewmodel[J]. Apress, 2011.
- [19]麦冬,陈涛,梁宗湾. 轻量级响应式框架 Vue.js 应用分析[J].信息与电脑(理论版). 2017(07).
- [20]张耀春.Vue.js 权威指南[M].电子工业出版社.2016.
- [21]徐鹏涛.基于 Vue 的前端开发框架的设计与实现[D].济南:山东大学,2020.
- [22]朱二华.基于 Vue.js 的 Web 前端应用研究[J].科技与创新.2017(20):119-121.
- [23]李洁.主流 JavaScript 框架-AngularJS,React 和 Vue 使用体会 [J].电脑迷,2019(01):77.
- [24]Youseef Alotaibi, Fei Liu. A novel secure business process modeling approach and its impact on business performance[J]. Information Sciences: An International Journal, 2014, 277:375-395.
- [25]焦鹏琿.基于 SpringBoot 和 Vue 框架的电子招投标系统的设计与实现[D].南京大学,2018.
- [26]王志文.Vue+Elementui+Echarts 在项目管理平台中的应用[J]. 山西科技,2020,35(06):45-47.
- [27]徐鹏涛.基于 Vue 的前端开发框架的设计与实现[D].山东大学,2020.
- [28]朱道远,郑胜,曾祥云,徐高贵.基于 JSON 的 Ajax 数据交换技术及应用[J].电脑编程技巧与维护,2015,4(16):14-15+32.
- [29]孟祥双,前后端分离式 WEB 应用开发研究[J].电子元器件与信息技术 2019.3(06):40-43.
- [30]唐汉明,翟振兴,关宝军.深入浅出 MySQL 数据库开发、优化与管理维护.人民邮电出版社,2014.1.
- [31]蔡祎炜.面向内存数据库的混合内存系统模拟与性能分析[D].哈尔滨工业大学, 2018.
- [32]张鹏飞,王乾,胡晓冬.基于 Node.js 和 JS 的前后端分离实现[J].计算机软件,2019,40(04):11-17.

- [33]路雯雯. 支持前后端分离的 JavaScript 开发框架的研究及在内容管理系统中的应用[D].山东大学,2017.
- [34]李兵.高职院校教师绩效考核系统的设计与实现[J].中小企业管理与科技(中旬刊),2015,03:283-284.
- [35]陈晨,杨亚彬. 基于 ASP.NET 的员工绩效考核系统的设计与实现[J].微型电脑应用,2015,02:28-30.
- [36]白兰芬.基于微服务架构的高校年度绩效考核系统的设计与实现[D].西安:西安电子科技大学 2020.
- [37]陈瑶.基于 Node.js 高并发 web 系统的研究与应用[D].成都:电子科技大学, 2014.
- [38]张旭超 .洛阳农信社绩效系统的设计与实现 [J].电子科技大学学报,2016,(12):77-78.
- [39]贺紫珺.基于 SpringBoot 和 Vue 框架的第三方医疗器械供应链平台的设计与实现[D].东华大学,2019.
- [40]何城政.基于 Web 的技师院校教师信息综合管理系统的开发构想[J].电子技术与软件工程.2015(05).
- [41]罗潇潇. 基于 B/S 结构的密封企业项目管理系统设计与实现[D].电子科技大学, 2020.
- [42]颜治平.基于 SpringBoot 和 Vue 框架的教代会提案系统的设计与实现[J].科技创新与应用,2020(03):91-93+95.
- [43]Kim, Kangseok, Ha, Wonjeong, Choi, Okkyung et al. An interactive pervasive whiteboard based on MVC architecture for ubiquitous collaboration[J]. Multimedia tools and applications, 2015, 74 (5) :1557-1576.
- [44]汪云飞. JavaEE 开发的颠覆者:Spring Boot 实战[M].电子工业出版社, 2016.
- [45]邓丽萍,黑盒测试之判定表法[J].福建电脑,2016,32(10):79-79,96.
- [46]武昭字,张月琴,阎华等.软件测试方法的研究与应用[J].太原理工大学学报,2016,47(03)379-383.