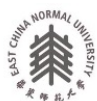


分类号: \_\_\_\_\_

密 级: \_\_\_\_\_

学校代码: 10269

学 号: 71194501266



華東師範大學

East China Normal University

硕士专业学位论文

MASTER'S DISSERTATION

# 论文题目: 基于 SpringBoot 和 Vue 的全能型供电所管理系统设计与实 现

院 系: 软件工程学院

专业学位类别: 工程硕士

专业学位领域: 软件工程

学位申请人: 朱 静

指导教师: 李成举 教授

2024 年 4 月 22 日

Dissertation for master degree in 2024

University code:10269

Student ID: 71194501266

# East China Normal University

**Title: Design and implementation of all-round power supply station management system based on SpringBoot and Vue**

Department :	<u>Software Engineering Institute</u>
Type :	<u>Master of Engineering</u>
Domain :	<u>Software Engineering</u>
Candidate :	<u>Jing Zhu</u>
Supervisor :	<u>Professor Chengju Li</u>

April 22,2024

## 朱静 硕士学位论文答辩委员会成员名单

姓名	职称	单位	备注
曾鹏	教授	华东师范大学	主席
刘虹	研究员	华东师范大学	
洪璇	副教授	上海师范大学	

## 摘 要

电力作为重要能源的基础，影响着人民日常生活和经济发展的方方面面。随着社会数字化、智能化的不断更新迭代，人民生活水平的不断提高，人们对供电服务质量和供电管理水平要求越来越高。供电所是国家电网公司最基层的一线阵地和窗口，目前供电所管理存在系统分散、数据壁垒、业务融合度低等问题。面对逐渐复杂的指标和不断更新的工作要求，传统的管理模式已不能满足当前的工作需要，且目前存在重业务轻管理的问题，不利于供电所综合管理能力的提升。

本文针对供电所管理存在以上一系列的问题和要求，通过对国内外供电所管理进行研究分析，设计并实现了全能型供电所管理系统。系统基于 B/S 架构，运用 MVC 设计模式，采用前后端分离技术。前端使用 Vue 框架整合 Element-UI 框架，后端使用 SpringBoot 框架。结合 Echarts 库，提供可视化数据展示，并对页面性能加以优化。采用 AOP 技术，提高代码的复用性和可维护性。使用 MySQL 和 Redis 数据库交互操作的形式，加快访问速度。系统经过功能性和非功能性测试，各方面表现良好，符合设计要求。

与传统供电所管理模式相比，针对其中存在的管理水平参差不齐和系统分散问题，本文研究设计并实现了基于 SpringBoot 和 Vue 的全能型供电所管理系统。该系统通过数据整合和数字化展示，为管理者提供清晰的数据分析和整改方案，从而提高管理效率，减轻基层工作负担。利用 TopN 算法和权重乘积之和算法实现智能派单，代替人工繁琐的派单步骤，有效提升数据质量及派单准确性。采用最小二乘法的用电预测算法，准确预测未来的用电趋势，帮助供电所和客户提前做好计划，优化电力资源的分配，增强了供电所的服务能力，也提高了客户的满意度，极大地提升了供电所的运营效率和服务质量。

**关键词：**全能型供电所，管理系统，SpringBoot 框架，Vue 框架，数字化

## ABSTRACT

Electricity as the important energy, affects all aspects of people's daily life and economic development. As the digitalization and intelligence of society continue to update, people's living standards continue to improve, people's demand for power supply service quality and power supply management level is getting higher and higher. The power supply station as the basic unit of the State Grid Corporation of China, the current management of the power supply station suffers from fragmented systems, data barriers, and low business integration. With the increasingly complex indicators and constantly updated work requirements, the traditional management model can not meet current work needs. And there is currently the problem of focusing on business but not management, which is not conducive to the improvement of the comprehensive management capacity of the power supply stations.

In this paper, in view of the above series of problems and requirements in the management of the power supply stations, through research and analysis of the management of power supply stations at home and abroad, design and implement an all-round power supply station management system. The system adopts B/S architecture, uses MVC design pattern, the separation of front-end and back-end. The front-end uses the Vue framework to integrate the Element-UI framework, and the back-end uses the SpringBoot framework. Combined with the Echarts library, it provides visual data display and optimizes page performance. Adopt AOP technology to improve code reusability and maintainability. Coordinate MySQL and Redis database to speed up access. The system has undergone functional and non-functional testing, has performed well in all aspects. The system complies with the design requirements.

Compared with the traditional power supply stations management mode, for the problems of uneven management level and system dispersion that exist in it, this paper, design and implementation of all-round power supply station management system based on SpringBoot and Vue. Through data integration and digital display, the system provides managers with clear data analysis and rectification plans, thus improving management efficiency and reducing the workload of the grassroots. Intelligent dispatching using TopN Algorithm and Weighted Sum Algorithm

replaces the tedious manual dispatching steps, effectively improving data quality and dispatching accuracy. Adopting the Least Squares Method power consumption prediction algorithm, it accurately predicts future power consumption trends, helps power supply stations and customers plan ahead, optimises the allocation of power resources, enhances the service capability of power supply stations, and also improves customer satisfaction, which greatly improves the operational efficiency and service quality of power supply stations.

**Keywords:** *All-round power supply station, management system, SpringBoot framework, Vue framework, digitalization*

# 目 录

摘 要.....	I
ABSTRACT.....	II
目 录.....	IV
图目录.....	VIII
表目录.....	XI
第一章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 研究现状.....	2
1.2.1 国内研究现状 .....	2
1.2.2 国外研究现状 .....	5
1.2.3 前端框架研究现状 .....	6
1.2.4 后端框架研究现状 .....	6
1.3 研究内容.....	7
1.4 论文的组织结构.....	8
第二章 关键技术概述 .....	10
2.1 B/S 架构.....	10
2.2 MVC 模式 .....	11
2.3 SpringBoot 框架 .....	12
2.4 Vue 框架 .....	14
2.5 Element-UI 框架.....	15
2.6 Echarts 库.....	15
2.7 Shiro 安全机制.....	16
2.8 Redis 数据库 .....	16
2.9 MyBatis 框架.....	16
2.10 排序算法.....	17
2.11 权重算法.....	17

2.12 拟合算法.....	17
2.13 本章小结.....	18
<b>第三章 全能型供电所管理系统需求分析 .....</b>	<b>19</b>
3.1 需求分析概述.....	19
3.2 用户需求分析.....	19
3.3 功能性需求分析.....	20
3.3.1 系统管理模块 .....	21
3.3.2 工作看板模块 .....	22
3.3.3 指标看板模块 .....	23
3.3.4 营销服务模块 .....	24
3.3.5 核算管理模块 .....	24
3.3.6 工单管理模块 .....	25
3.3.7 安全生产模块 .....	26
3.3.8 综合管理模块 .....	27
3.4 非功能性需求分析.....	27
3.5 可行性分析.....	28
3.6 本章小结.....	29
<b>第四章 全能型供电所管理系统详细设计 .....</b>	<b>30</b>
4.1 系统总体架构设计.....	30
4.2 系统功能模块设计.....	31
4.2.1 系统管理模块 .....	32
4.2.2 工作看板模块 .....	33
4.2.3 指标看板模块 .....	34
4.2.4 营销服务模块 .....	34
4.2.5 核算管理模块 .....	34
4.2.6 工单管理模块 .....	35
4.2.7 安全生产模块 .....	36



4.2.8 综合管理模块 .....	36
4.3 数据库设计 .....	37
4.3.1 数据库概念设计 .....	37
4.3.2 数据库表设计 .....	39
4.4 算法设计 .....	44
4.4.1 基于 TopN 和权重乘积之和的自动派单算法 .....	45
4.4.2 基于最小二乘法的用电量预测算法 .....	46
4.5 本章小结 .....	47
<b>第五章 全能型供电所管理系统实现 .....</b>	<b>48</b>
5.1 全能型供电所管理系统功能模块实现 .....	48
5.1.1 系统管理模块 .....	48
5.1.2 工作看板模块 .....	50
5.1.3 指标看板模块 .....	53
5.1.4 营销服务模块 .....	54
5.1.5 核算管理模块 .....	55
5.1.6 工单管理模块 .....	57
5.1.7 安全生产模块 .....	58
5.1.8 综合管理模块 .....	59
5.2 辅助全能型供电所管理系统功能模块实现 .....	59
5.2.1 登录注册模块 .....	60
5.2.2 菜单界面模块 .....	63
5.3 算法实现 .....	64
5.3.1 基于 TopN 和权重乘积之和的自动派单算法 .....	64
5.3.2 基于最小二乘法的用电量预测算法 .....	67
5.4 本章小结 .....	69
<b>第六章 系统测试 .....</b>	<b>70</b>
6.1 系统测试环境 .....	70

6.2 功能性测试.....	70
6.2.1 全能型供电所管理系统功能模块测试 .....	71
6.2.2 辅助全能型供电所管理系统功能模块测试 .....	73
6.3 单元测试.....	73
6.4 非功能性测试.....	74
6.4.1 性能测试 .....	74
6.4.2 其他非功能性测试 .....	77
6.5 系统测试结果分析.....	77
6.6 本章小结.....	77
第七章 总结与展望 .....	78
7.1 总结.....	78
7.2 展望.....	79
参考文献.....	80
致    谢.....	85

## 图目录

图 1-1 论文结构图 .....	8
图 2-1 B/S 架构图 .....	10
图 2-2 MVC 架构图 .....	11
图 2-3 Spring 架构图 .....	13
图 2-4 Vue 架构图 .....	14
图 3-1 系统总体功能用例图 .....	21
图 3-2 系统管理模块用例图 .....	22
图 3-3 工作看板模块用例图 .....	23
图 3-4 指标看板模块用例图 .....	23
图 3-5 营销服务模块用例图 .....	24
图 3-6 核算管理模块用例图 .....	25
图 3-7 工单管理模块用例图 .....	26
图 3-8 安全生产模块用例图 .....	26
图 3-9 综合管理模块用例图 .....	27
图 4-1 系统架构图 .....	30
图 4-2 系统技术架构图 .....	31
图 4-3 系统功能模块图 .....	32
图 4-4 工作看板模块操作流程图 .....	33
图 4-5 核算管理模块的操作流程泳道图 .....	34
图 4-6 工单管理模块操作流程时序图 .....	35
图 4-7 安全生产模块的操作流程泳道图 .....	36
图 4-8 综合管理模块操作流程时序图 .....	37
图 4-9 用户角度 E-R .....	38
图 4-10 公司管理员角度 E-R 图 .....	38
图 4-11 自动派单流程图 .....	45
图 4-12 预测客户用电量流程图 .....	46

图 5-1 部门管理模块界面 .....	48
图 5-2 角色管理模块界面 .....	49
图 5-3 菜单管理界面 .....	49
图 5-4 Controller 层菜单管理关键代码 .....	50
图 5-5 公司管理员工作看板界面 .....	50
图 5-6 调取 Json 分布图坐标数据关键代码 .....	51
图 5-7 鼠标悬停在海河供电所上展示界面 .....	51
图 5-8 所长工作看板界面 .....	52
图 5-9 核算员工作看板 .....	52
图 5-10 Rolecontroller 类中关键代码 .....	53
图 5-11 Usercontroller 类中关键代码 .....	53
图 5-12 指标看板模块界面 .....	54
图 5-13 台区线损合格率界面 .....	54
图 5-14 台区线损汇总界面 .....	55
图 5-15 台区线损汇总关键代码 .....	55
图 5-16 欠费明细界面 .....	56
图 5-17 导出电费明细关键代码 .....	56
图 5-18 生成工单关键代码 .....	57
图 5-19 工单管理模块界面 .....	57
图 5-20 AOP 模式拦截请求关键代码 .....	58
图 5-21 故障抢修模块界面 .....	58
图 5-22 所务公开模块界面 .....	59
图 5-23 员工管理模块界面 .....	59
图 5-24 注册界面 .....	60
图 5-25 MD5 算法加密关键代码 .....	60
图 5-26 登录界面 .....	61
图 5-27 记住密码功能关键代码 .....	61

图 5-28 Axios 请求拦截器的关键代码 .....	62
图 5-29 Axios 响应拦截器的关键代码 .....	62
图 5-30 公司管理员菜单界面 .....	63
图 5-31 所长菜单界面 .....	63
图 5-32 菜单折叠前界面 .....	63
图 5-33 菜单折叠后界面 .....	63
图 5-34 菜单折叠功能关键代码 .....	64
图 5-35 实时查看接单情况界面 .....	66
图 5-36 xy散点图及线性拟合图 .....	68
图 5-37 实验对比图 .....	68
图 6-1 系统并发测试图 .....	75
图 6-2 系统吞吐量图 .....	75
图 6-3 系统响应时间图 .....	76
图 6-4 系统响应时间分布图 .....	76

## 表目录

表 4-1 台区信息表 .....	39
表 4-2 台区线损表 .....	40
表 4-3 台区线损汇总表 .....	40
表 4-4 客户信息表 .....	41
表 4-5 电费缴纳表 .....	42
表 4-6 电费回收率 .....	42
表 4-7 工单表 .....	43
表 4-8 客户报修表 .....	44
表 5-1 详细参数表 .....	65
表 5-2 5000 户居民客户用电量记录表 .....	67
表 6-1 全能型供电所管理系统工单管理模块测试用例表 .....	71
表 6-2 全能型供电所管理系统核算管理模块测试用例表 .....	72
表 6-3 全能型供电所管理系统综合管理模块测试用例表 .....	72
表 6-4 全能型供电所管理系统辅助功能模块测试用例表 .....	73
表 6-5 单元测试覆盖率统计表 .....	74
表 6-6 全能型供电所管理系统的其他非功能性测试用例表 .....	77

# 第一章 绪论

## 1.1 研究背景及意义

随着我国市场经济的持续发展，电力体制改革步伐日益加快，供电企业也在不断地进行改革<sup>[1]</sup>。供电企业作为企业生产经营和人民日常生活的服务型企业，拥有独特的电力行业属性和社会责任。然而，伴随我国发展进入新常态，人民生活水平日益提高，对供电企业员工的服务工作提出了更高的要求<sup>[2]</sup>。

供电所作为国家电网最基层的一线作业和供电服务单位，是供电企业业务执行的最小单元，承担着服务地方经济社会发展，密切联系人民群众和地方政府的重要职责<sup>[3]</sup>，扎实推进村网共建、乡村振兴等中国式现代化战略部署落地实施<sup>[4]</sup>。供电所的管理水平和服务质量直接影响着供电企业的社会形象。

国家电网自 2017 年以来全面开展全能型供电所建设工作，目的是改善生产条件，推进营配合一，积极打造业务协同运行、人员一专多能、服务一次性到位的全能型供电所<sup>[5]</sup>，旨在以客户为中心，构建执行有力、响应快速、反应敏捷的新型服务模式，不断提升乡镇供电服务质量和效率<sup>[6]</sup>，践行“人民电业为人民”的企业宗旨。

本课题来源于盐城供电公司，盐城供电公司现有 98 个供电所，分布在区县各个乡镇。目前，各供电所整体管理水平参差不齐，普遍存在服务半径过大、基础设施薄弱、人力资源不足等难题，已成为建设全能型供电所的短板<sup>[7]</sup>。

本文根据实际工作，结合全能型供电所建设工作中存在的问题开展研究，研究意义如下：

（1）推动数字所务建设。供电所智能化管理程度不足，大部分供电所所务公开仍以手工记录为主，记录不规范，少记、漏记情况时有发生，存在一定的管理漏洞。实现供电所无纸化所务管理，结合所务公开页面，对供电所所务线上管理，增强工作透明性，解决所务管理费时费力的问题。

（2）提升指标管理水平。由于所内综合管理人才少，效率低，异常指标的监控不到位，存在一定的管理风险。将各项指标以图形和数据的形式进行可视化

展示，综合展现各供电所指标在全市、全区县的综合排名，对比全市平均值最优值，易于同业对标学习，调整工作计划，提高各专业重点指标。

（3）推进智能自动派单。供电所管理辖区面积广，管理客户数量多，用户分布广泛，客户类型多元化，对供电所员工进行科学高效地管理尤为重要。利用数字化技术赋能关键业务，汇集供电所九大类业务工单，以工单形式智能自动派发工作任务，代替了人工繁琐的派单步骤，避免了人工录入出错，有效提升了数据质量及派单准确性。

（4）助力“双碳”目标实现。随着我国电力资源的持续短缺，必须加强电力资源的有效利用，加快实现“双碳”的目标。通过实时关注客户的用电量趋势，使用精准的数据分析预测其未来用电量，及时告知电力峰值，帮助客户减少用电成本，提高客户满意度，有效推进节能减排。

（5）实现基层减负赋能。供电所作为最基层单位，涉及各个专业的应用系统，数量多，范围广，各专业之间的系统存在壁垒，数据不贯通，业务不融合。通过优化整合各专业业务，数字化展现工作看板。同时基于各专业的数据融合分析，完成数字化转型，有利于供电所管理人员快速上手并使用，有效提高了工作效率。

## 1.2 研究现状

### 1.2.1 国内研究现状

电力行业在国民经济与社会发展中一直处于支柱产业地位，自中华人民共和国成立以来，党中央和国务院考虑国内外经济发展的需要，重视电力行业的改革与发展，电力管理体制先后经历了九次变革，电力行业迅速发展，取得巨大成就，为国家经济的快速发展提供有力的支撑<sup>[8]</sup>。1998年，党中央和国务院根据国内外经济发展形势，实施“两改一同价”政策，实现城乡同网同价，加快了乡镇电力体制改革<sup>[9]</sup>。2002年，中国国家电力公司拆分重组成11家公司：国家电网公司和中国南方电网公司，五家发电公司以及四个辅业公司<sup>[10]</sup>。2005年，国家电网公司部署“创一流同业对标”工作<sup>[11]</sup>，于2010年建立同业对标常态工作机制。



将“创一流”作为公司目标，确立追赶标杆，建立竞争优势，从而提升整体管理水平。立足于行业领先的地位，全面查找差距，明确目标，制定相应的对策并付诸实践，具有一定的可操作性<sup>[12]</sup>。崔梅雪分析并研究了同业对标的应用对供电所专业管理水平产生积极作用<sup>[13]</sup>，供电所的管理水平直接影响到供电服务质量。目前乡镇供电所普遍面临着电费回收、安全生产和指标管控等压力，必须要不断提高供电所员工的业务水平和服务沟通能力，充分调动员工的积极性和主动性，助推全能型供电所建设工作的顺利开展。

盐城供电公司于 2010 年启动了供电所同业对标模式，在各个供电所业务水平进行对比，不断收集特色管理经验并加以改进和推广，使盐城辖区供电所业务水平在全省范围中名列前茅，激发了员工不断追求卓越的热情与动力。随着供电所工作要求的日益复杂，各种指标数据呈几何级数的增长趋势，基于当前工作要求的变化，传统管理模式已不再提供充分的支持。更多的重视放在业务发展上，而对管理层面的关注则相对较少，不利于提高供电所管理效率和质量，亟需开发与当前工作实际相适应的管理系统，将传统的管理方式与数字化相结合，更加符合国家电网的企业目标。与全市乃至全省优秀的供电所进行对比，找出自身管理上存在的问题，并采取相应的改进措施和方法，提高供电所人员的业务素质与服务意识。

国务院于 2015 年发布了《关于进一步深化电力体制改革的若干意见》，确立了“放开两头、管住中间”的改革思路<sup>[14]</sup>，标志着新一轮电力体制改革的全面启动。在售电侧引入社会资本，形成多样化的售电主体竞争格局<sup>[15]</sup>。随着售电市场的放开，国家电网经营管理观念逐步改变，要确保稳定的收入，就必须不断降低成本。在开放的售电市场中，通过节约成本并提升盈利能力，不断提高供电所人员的工作效率和业务水平成为必要条件。然而，当前又面临着供电所合并和严重人员短缺的问题，售电市场的多样化竞争环境将导致重要客户流失和市场占有率降低。

“十四五”期间，国家电网董事长辛保安提出“一体四翼”发展布局，加快建设具有中国特色国际领先的能源互联网企业的目标<sup>[16]</sup>，为全面建设社会主义

现代化国家作出积极贡献。党的十九大提出建设“数字中国”的重大战略命题，党的二十大进一步强调“加快建设数字中国”，数字化已经成为经济社会增长和企业创新发展的重要动力<sup>[17]</sup>。目前国家电网高度重视数字化转型，积极打造数智化坚强电网，加快构建新型电力系统<sup>[18]</sup>。

龙兰英等学者阐述了深入研究乡镇供电所的管理模式是电力企业必须开展的一项重要工作，是改善乡镇供电所供电服务质量和促进乡镇经济发展的必由之路<sup>[19]</sup>。国网山东省电力公司莱西供电公司日庄供电所邢宪蕊等人通过创新管理模式，构建“党建+新绩效”新机制，解决因结构化缺员问题导致的供电所服务困境，不断激发员工的干事热情，推动供电所业绩持续提升<sup>[20]</sup>。国网福建电力公司龙岩供电公司李羨才等人致力于提升供电所的基础管理工作，将“数字赋能、基层减负”作为策略切入点，创新性地提出一种基于“中央厨房式”的管理模式，通过将乡镇供电所的任务清单数字化，形成一个“中央厨房”数据看板，以此实时监控各项工作任务完成情况及存在问题，实现为一线人员减负，提升运营管理质效<sup>[21]</sup>。国网福建电力公司上杭供电公司古田供电所吴祖培等人以建设国家电网数字化所级示范点为目标，开发了一款日线损异常台区监测的RPA（Robotic Process Automation）机器人，旨在替代供电所员工进行线损监测和精准管控，提升工作效率，减少员工的工作负担<sup>[22]</sup>。盐城三新供电服务有限公司大丰分公司李刚提出推行供电所网格化管理的策略，进一步优化农村供电服务模式及提升供电服务质效，推进全能型乡镇供电所建设工作<sup>[23]</sup>。

国内对供电所的管理方式都采取了一系列的创新，能够有效提高工作效率，保障客户日常生产和生活用电。一是结合数字化技术开发RPA机器人，减少重复性工作。二是结合数字化技术，实时掌握工作完成情况和存在问题，为基层减负赋能。三是推行供电所网格化管理，提高工作效率和服务质量。四是构建“党建+新绩效”新机制，将“党建+”融入柔性网格绩效团队建设中，提升工作业绩指标。因此在供电所管理中，推动数字所务建设，提升指标管理水平，实现智能化自动派单，实时预测客户用电量，减少人力物力，为基层减负增效，对于稳定供电公司收入和增强竞争能力方面具有十分关键的作用。

### 1.2.2 国外研究现状

对于国外乡镇供电方式的相关问题，Muttaqi和Ledwich对国外乡镇供电方式的改进与服务战略进行了研究<sup>[24]</sup>，Singh对印度的绿色电力供给方式进行探讨<sup>[25]</sup>，Harish研究了光伏电池技术在乡镇供电中应用的可行性，并对节约成本的问题作进一步分析<sup>[26]</sup>。随着发达国家电力市场开放自由竞争模式的加剧，供电企业之间的竞争关系逐渐显现出来，在乡镇供电机构的管理中，提高客户服务水平越来越受到重视。

一是美国乡镇供电合作社的运营方式。自罗斯福新政以来，美国乡镇电气化逐渐发展，通过优惠政策鼓励乡镇供电合作社的发展。二是法国和英国之间提供开放自由竞争模式。与美国经营方式相似，国家对乡镇供电提供了大量的政策扶持，法国将乡镇供电建设纳入到国家发展规划中。以法国为例，法国将乡镇供电运营维护与电力购销分离，POWEO公司只负责电力购销，并非电力资产的所有者。英国公司垄断输电侧，为配电、发电和售电公司创造了竞争条件，由专业的公司负责经营和管理乡镇电力供应。三是澳大利亚的福利政策。澳大利亚通过实施福利政策推动乡镇电力供应的发展，从法律上保障乡镇电力客户获得廉价电力的权利，并通过在城乡之间交叉补贴分摊国家电力供应成本。

在国外，各国都采取了一系列相应的措施，引导供电机构的管理促进乡镇供电发展。一是针对乡镇建立专业的供电管理机构，实施统一化管理策略，确保乡镇供电的统一管理和可靠运行。二是制定完善的电力扶持政策，各国制定了相应的法律、制度等措施，维护电力市场的稳定。三是对乡镇供电机构进行全面评估，一些国家甚至对其进行了统一，美国实现规范化管理，提升设备运行维护以及检修工作的效率，确保电力供应的可靠性。

鉴于不同国家之间存在显著的差异，我国的电力供应体系表现出更高的复杂性、密集性和多样性。因此，乡镇供电所作为国家电网的最末端，更加需要结合数字化转型，开发一个符合当前工作实际的管理系统，实现基层减负和提质增效，全面提升供电所的服务效率和客户的满意度。

### 1.2.3 前端框架研究现状

目前,前端技术领域正呈现出多元化、精细化、跨平台的发展趋势,最流行的三个前端开发框架分别是React、Angular和Vue<sup>[27]</sup>。

React因其卓越的性能表现和组件化的开发范式而备受赞誉。通过创新的虚拟DOM技术,React更加智能地更新页面,避免不必要的重渲染,从而显著提升了用户体验。此外,React的组件化开发方法使得代码结构更加清晰、模块化,极大地增强了代码的可维护性和可扩展性。

Angular是一个功能全面的前端框架,支持数据绑定和依赖注入机制,显著提升了代码的健壮性和可维护性。Angular还提供了丰富的模板语法和指令集,使开发者能够更高效地构建用户界面。Angular特别注重测试和性能优化,内置了完善的测试工具和性能分析工具。

Vue的语法直观明了,因其简单易学和渐进式开发的特性而备受欢迎。同时,Vue支持组件化开发模式,提供丰富的API和指令,极大提高了开发过程的便捷性。Vue还强调轻量级和灵活性,可以根据项目需求灵活选择使用的功能和工具。

综上所述,React、Angular和Vue各有其优势和特点。React适合构建大型复杂应用,Angular则适合构建企业级应用,而Vue适合快速开发和轻量级项目,因此本文采用Vue作为前端开发框架。

### 1.2.4 后端框架研究现状

Spring是一个基于Java的开源框架,它在Java开发社区中拥有非常高的市场占有率,其推出的Spring MVC框架被广泛的应用于各类Web产品的研发当中<sup>[28]</sup>。Spring先后发布了多个版本,随着版本的更新,繁琐的配置工作在逐步减少。

Spring Boot项目是Pivotal团队开发的一个全新Spring框架,旨在解决Spring开发过程中复杂的配置问题,并朝着零配置的目标迈进。Spring Boot极大地简化了应用开发的搭建过程,使开发人员能够摆脱繁琐的配置文件,将更多精力投入到项目逻辑的研发上。Spring Boot致力于成为快速应用开发领域的引领者,通过提供简洁、快速且易于配置的开发体验,帮助开发者更高效地构建和部署应用。

### 1.3 研究内容

本人目前在乡镇供电所工作，是一名台区经理。主要负责 4 个台区，涉及 2 个小区，约 600 名电力客户的电费回收、计量采集维护等工作。除本职工作外，还需协助各专业管理人员完成相应工作。工作中经常使用多个系统，范围广，各专业之间的系统存在壁垒，业务不融合，数据不贯通。供电所综合管理人才少，效率低，异常指标的监控不到位，存在一定的管理风险。于是，本人基于各专业的数据融合分析，横向业务贯穿，覆盖日常工作内容，学习国内外相关领域的优秀技术和成果，研究设计一套适用于供电所日常工作的管理系统，实现供电所核心业务线上融合，关键指标在线监测，助推基层减负增效。

本文主要研究内容如下：

（1）在日常工作中，现场调研各管理专业现状，收集相关工作指标，结合实际工作，从技术、经济和系统运行三个方面进行可行性分析，从用户、功能性和非功能性开展需求分析。

（2）实现供电所无纸化所务管理。结合所务公开页面，对供电所所务线上管理，增强工作透明性，解决所务管理费时费力的问题。

（3）对各项指标进行统计分析，通过图形和数据的形式更直观的呈现数据。为加强同业对标学习，综合展示供电所各项指标在全市、全区县的综合排名，对比全市平均值最优值，及时调整工作重点，迅速提高关键指标。

（4）汇集供电所九大类业务工单，基于 TopN 和权重乘积之和的自动派单算法，代替了人工繁琐的派单步骤，避免了人工录入出错，有效提升了数据质量及派单准确性。

（5）通过优化整合各管理专业业务，数字化展现工作看板。基于各专业的数据进行融合分析，完成数字化转型，有利于供电所管理人员快速上手并日常使用，有效提高了工作效率。

（6）基于最小二乘法的用电量预测算法构建线性回归模型，准确预测未来的用电趋势。帮助供电所和客户提前做好计划，优化电力资源的分配，增强了供

电所的服务能力，也提高了客户的满意度，极大地提升了供电所的运营效率和服务质量。

(7) 对系统进行功能性和非功能性测试，验证测试结果与预期结果是否一致，确保系统的稳定性。

## 1.4 论文的组织结构

本文深入研究了系统中使用的前端和后端框架及其他框架技术，并详细描述了每个功能模块的设计和实现。本文共分为七个章节，论文结构图如图 1-1 所示：

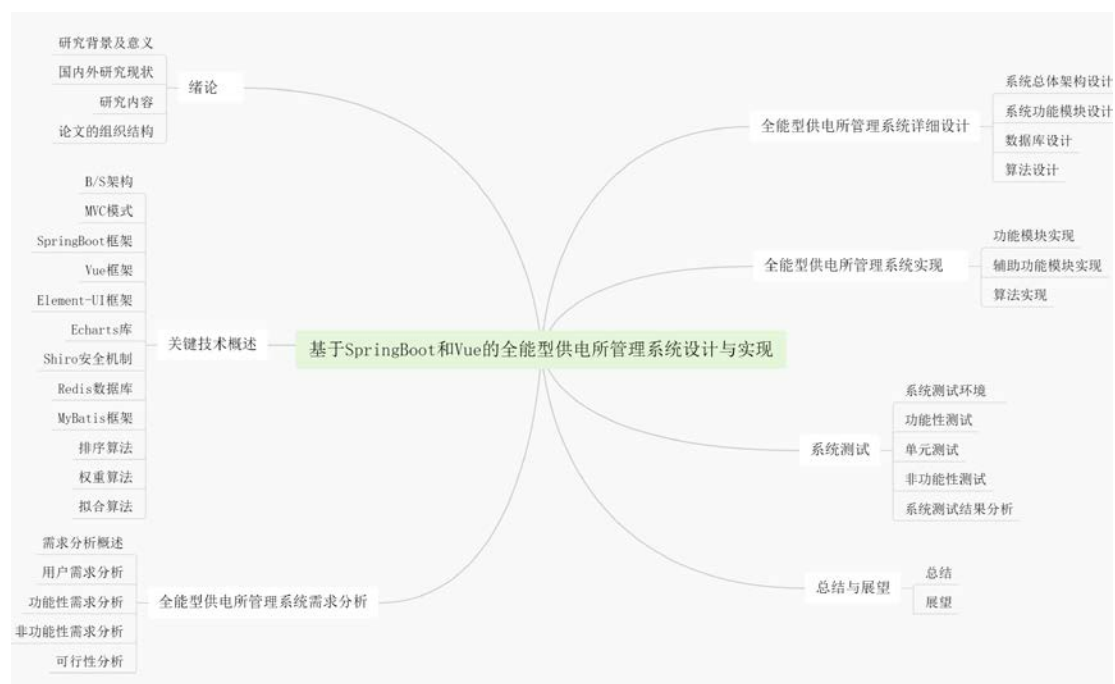


图 1-1 论文结构图

详细结构内容如下：

第一章：详细介绍本系统的开发背景和意义，对国内外相关研究和前后端框架技术现状进行了调研，提出论文的研究内容，规划论文的组织结构。

第二章：阐述了系统实现过程中的框架技术和算法，包括相关框架技术介绍和其他同类框架技术的对比。

第三章：整理并分析了系统在总体、功能性和非功能性等方面的需求。在功能性需求中，对功能模块进行了说明和用例图设计。在非功能性需求中，从系统性能、安全性、兼容性和易用性等方面考虑，介绍了系统的其他需求。

第四章：对系统进行详细设计。介绍了系统总体设计思路，划分系统中各个功能模块如工作看板模块、指标看板模块、营销服务模块、核算管理模块、工单管理模块、安全生产模块、综合管理模块和系统管理模块等，通过架构图与文字说明相结合的方式，详细介绍系统中各个功能模块的设计，以及数据库概念设计、数据库表设计和关键算法设计。

第五章：在系统需求分析与详细设计的基础上，将系统按模块划分，详细阐述实现细节，对重要功能界面进行展示，对关键算法进行介绍，对关键功能的部分代码进行描述。

第六章：对系统有代表性的功能进行黑盒和白盒测试，叙述了系统的测试过程，包括测试用例的设计、测试环境以及相应测试结果，不断对系统进行完善。

第七章：总结论文主要内容，就全能型供电所管理系统的未来扩展作进一步展望，根据供电所管理的发展趋势，提出后续将扩充的功能和数据。

## 第二章 关键技术概述

本章对全能型供电所管理系统开发过程中使用到的一些关键技术进行介绍，分析其优缺点，为本系统的开发提供理论参考和技术支持。系统采用 B/S 架构，使用 MVC 设计模式，面向切面编程。其他关键技术主要有 SpringBoot 框架、Vue 框架、Mybatis 框架、排序算法、权重算法和拟合算法等。

### 2.1 B/S 架构

在当今的互联网时代，有两种常见的软件架构框架，即 C/S 架构和 B/S 架构。前者最初由 Borland 公司开发，后者是由微软公司推动其发展。

“C/S”是“Client/Server”的缩写，即“客户端/服务器”架构<sup>[29]</sup>。C/S 架构通过为客户端和服务端分配不同的任务，有效地减少了两端之间通信的成本。但 C/S 架构对客户端和服务器的硬件设备具有一定的要求。根据不同的应用软件需求，客户端和服务端均需安装和配置特定的软件环境。在软件升级过程中，为保证软件的兼容性与性能，客户端也需要进行同步更新与安装。随着用户数量的增加，C/S 架构出现硬件成本、升级维护以及运行效率等一系列问题，导致无法满足部分用户的需求。

“B/S”是“Browser/Server”的缩写，即“浏览器/服务器”架构。B/S 架构对 C/S 架构进行了升级和优化，其中的业务操作是通过浏览器与服务器之间的交互完成的。B/S 架构图如图 2-1 所示：

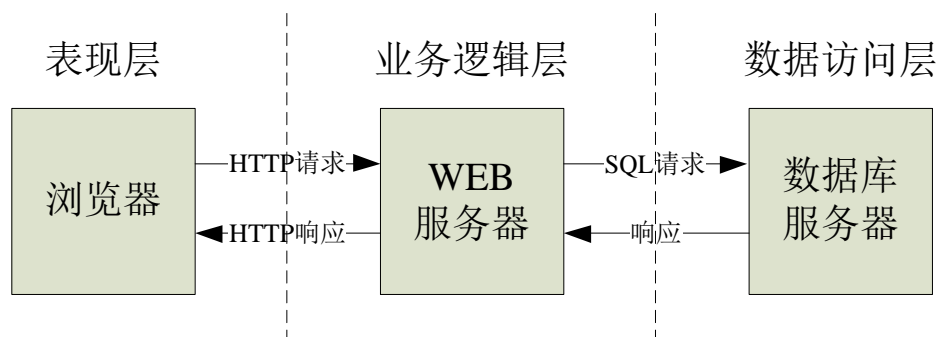


图 2-1 B/S 架构图



B/S 逻辑架构是由表现层、业务逻辑层和数据访问层组成<sup>[30]</sup>。表现层是指用户与系统进行交互的界面，比如用户使用浏览器进行查询和提交表单等操作。业务逻辑层的功能包括调用各功能程序，将表现层的数据请求发送到数据访问层。数据访问层负责对服务器数据进行读写操作，读取后数据返回给浏览器。

B/S 架构通过使用浏览器作为客户端，简化了客户端的部署和维护工作，使得用户无需安装特定的客户端软件即可访问应用程序。此外，这种架构模式利用了 Web 技术的通用性和互操作性，提高了系统的可扩展性和可访问性，这也是 B/S 架构在软件开发中得到越来越多应用的原因。

## 2.2 MVC 模式

MVC 模式是软件工程中的一种设计模式。在 MVC 模式中，M 是指 Model（模型），V 是指 View（视图），C 则是 Control（控制器）<sup>[31]</sup>，MVC 模式在 Model、View 和 Controller 之间建立了动态化的映射关系。使用 MVC 模式的目的是实现一种动态的程序设计，简化后续的扩展与修改。MVC 模式通过简化代码的复杂度，使程序的结构更加直观。MVC 架构图如图 2-2 所示：

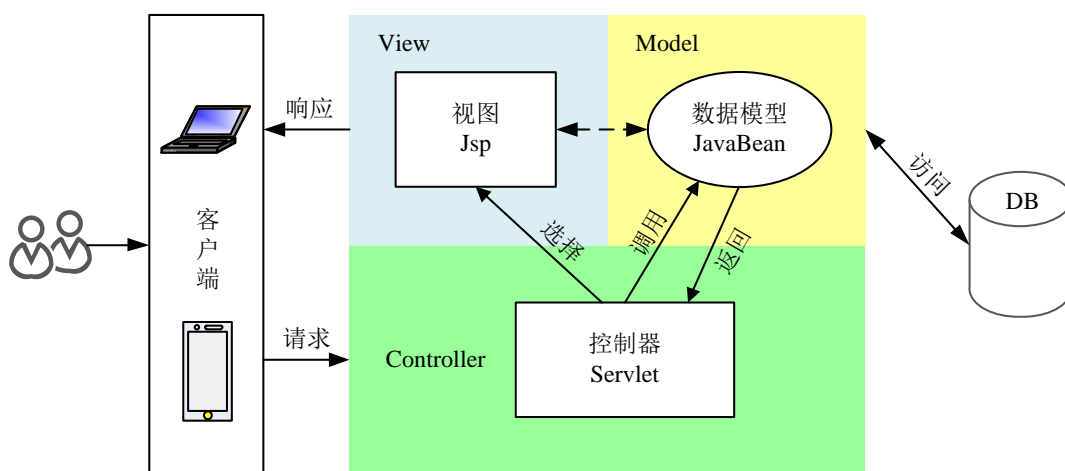


图 2-2 MVC 架构图

**模型层（Model）：**指从现实世界中抽象出来的对象模型。模型层封装了数据和对数据的操作，并与数据库进行交互，负责处理实际数据。

**视图层（View）：**视图层负责将客户端页面展示给用户，是客户端和用户之间的接口。

控制器（Controller）：控制器用于控制程序的流程，将模型中的数据展示到视图中。控制器承担了视图与模型之间交互的关键角色，具体包括控制用户输入的响应、响应方式以及交互流程。

## 2.3 SpringBoot 框架

SpringBoot 框架是由 Spring 框架改进而来的，了解 SpringBoot 框架前需要先认识 Spring 框架。

Spring 是一个功能强大而轻量级的开源框架，被广泛应用于 Java 企业级应用程序开发。与其他框架相比，Spring 框架简化开发步骤，具有高内聚低耦合的特点<sup>[32]</sup>。Spring 框架的主要优势之一就是分层架构，支持开发人员自主选择使用某一个组件。其本身又集成多种框架，具有较好的封装性，因此使用频率较高。

Spring 框架使得开发人员摒弃了早期的 EJB（Enterprise Java Bean）框架，其核心功能是 AOP（面向切面编程）和 IOC（控制反转）<sup>[33]</sup>，AOP 和 IOC 是现代 Java 应用程序开发中广泛使用的两种技术。在传统的 OOP（面向对象编程）中，通常将功能划分成多个对象，通过继承和组合来实现。使用 AOP 技术，将横向的功能从业务逻辑中分离，比如日志记录、性能监控、事务管理等功能，从而减少代码的冗余和重复。同时，AOP 技术能够提高代码的可重用性和可扩展性，因为它将横向的功能应用于多个对象上。IOC 技术是将对象的创建过程和依赖关系的处理交给外部容器实现，从而实现对象之间的解耦，提高代码的可维护性和可扩展性。本系统采用了其中的 AOP 模块，方便地实现对程序进行权限拦截和运行监控等功能。Spring 架构图如图 2-3 所示：

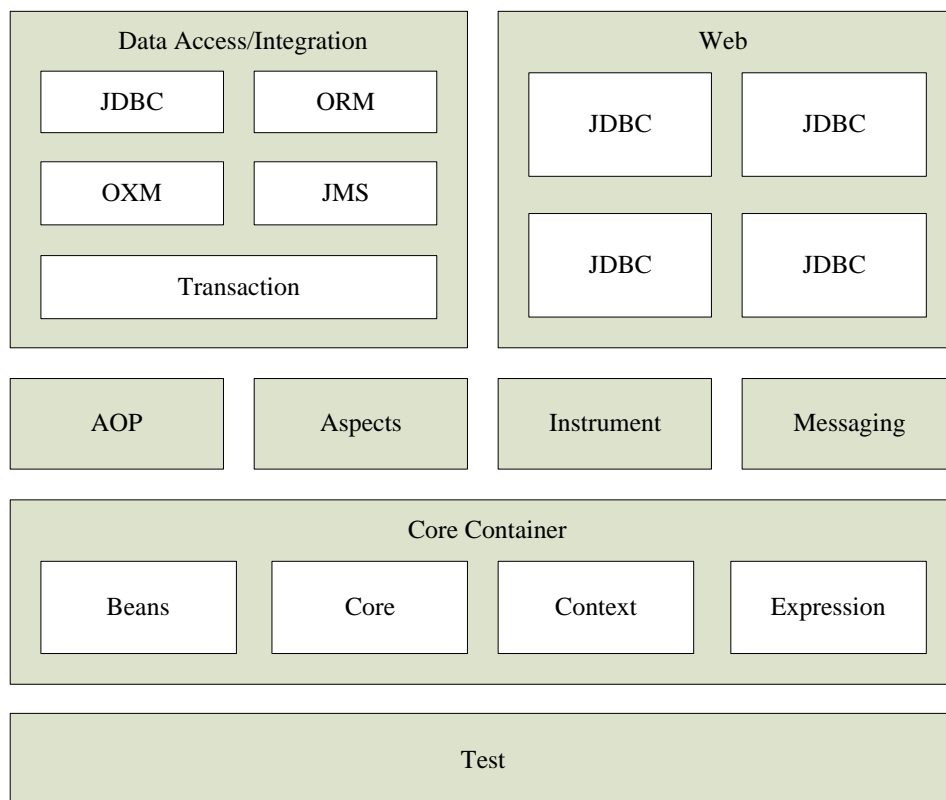


图 2-3 Spring 架构图

2013 年, Pivotal 团队开发出一个全新开源的 SpringBoot 框架, 简化了 Spring 框架各方面配置<sup>[34]</sup>, 目的是为 Spring 框架开发人员提供一个更快、更广泛的入门体验。Spring 框架的主要限制在于其对多种依赖配置的需求以及配置过程的复杂性, 严重降低 Java 开发人员的工作效率。而 SpringBoot 框架采用其固有的配置方法进行框架配置, 遵循“约定大于配置”的原则, 实现了自动装配的机制, 使开发人员不再使用复杂的定义方式进行配置。

SpringBoot 框架之所以受到很多开发人员的青睐, 是因为它还具有以下五个优点:

- (1) 不需要额外生成配置代码和 XML 配置文件<sup>[35]</sup>。
- (2) 使用内嵌的 Tomcat、Jetty 等服务器运行 SpringBoot 程序<sup>[36]</sup>。
- (3) 根据项目需求自动配置实体类和相应的初始值。
- (4) 提供大型项目的非业务性功能, 如安全管理, 运行数据监控等<sup>[37]</sup>。
- (5) 与其它框架良好的整合, 如 Vue、Mybatis 等。

## 2.4 Vue 框架

Vue.js（简称 Vue）是一款开源的 JavaScript 前端框架，侧重于视图层，专门设计用于构建用户界面。Vue 自底向上逐层开发，易于开发人员使用<sup>[38]</sup>。Vue 官方文档清晰，简单易学，容易与第三方库整合。Vue 框架中使用很多的其他技术，比如 HTML、CSS、NPM 等。Vue 架构图如图 2-4 所示：

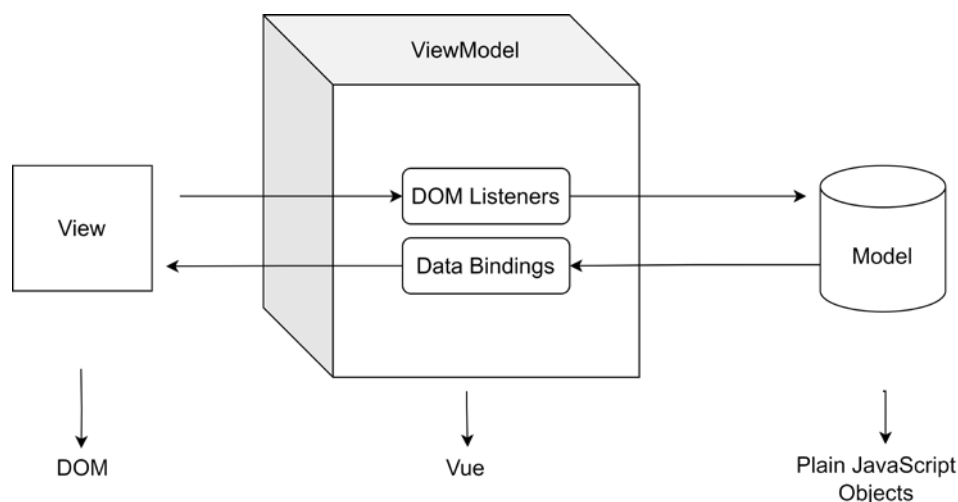


图 2-4 Vue 架构图

此外，Vue 框架还具有以下三个特点：

- （1）轻量级的框架。
- （2）响应式数据且支持数据双向绑定。
- （3）模块化开发模式。

目前，Web 前端有三大主流框架，分别是 Vue、React 和 Angular。三大框架都支持双向数据绑定，但是 Vue 与其他两个框架存在一些不同点。

Vue 并不直接支持旧版本的 IE 浏览器，尤其是 IE8 及以下版本，React 可以支持到 IE8 浏览器。React 采用虚拟 DOM 中的微操作实现对实际 DOM 的局部更新，Vue 采用异步批处理方式更新 DOM。但 React 不是一个完整的框架，还需要加上 ReactRouter 和 Flux 才能编写大型应用程序。Vue 的官方文档很清晰，而 Angular 学习成本高。因此，Vue 比 Angular 更加简单易学，使用场景比 Angular 更加灵活。

## 2.5 Element-UI 框架

Element-UI 框架是一个基于 Vue2.0 开发的前端框架，采用许多流行的 UI 设计和前端开发方式。它非常灵活且易于使用，集成一系列工具和组件，每个组件都有详细的实例，帮助开发人员快速构建出 Web 应用程序。Element-UI 包括按钮、表单、弹窗、菜单、导航等组件，这些组件都具有良好的自定义性和可扩展性，满足不同项目的需求。Element-UI 还提供丰富的主题和样式，让开发人员轻松地定制应用程序风格。Element-UI 是一个非常实用且强大的 UI 框架，大大提高 Web 应用程序的开发效率和质量。

Element-UI 框架具有响应式的设计，轻松地实现不同设备之间的适应性，并自动响应页面大小的变化。Element-UI 框架提供了一些方便的工具简化开发过程，如虚拟滚动、数据表格和表单验证等。

目前，Web 前端有四大主流 UI 框架，分别是 Element-UI、Ant Design、iView 和 LayUI 框架。Ant Design 在标准和视觉上更胜一筹，但发展时间短，说明文档中案例较少，技术上没有 Element-UI 成熟。iView 和 Element-UI 都基于 Vue 框架，实现数据的双向绑定，但 iView 在功能上没有 Element-UI 齐全。LayUI 组件齐全，界面美观，但部分组件基于 jQuery 框架，图标样式没有 Element-UI 齐全。

## 2.6 Echarts 库

ECharts 库是一个基于 JavaScript 框架的开源可视化前端图表库，支持在 PC 端和移动设备上流畅地运行，兼容市场上大多数浏览器。ECharts 包含多种实用性强的图标组件及图表类型，支持多种配合使用，且更加直观地显示，具有可交互、自定义数据等优点<sup>[39]</sup>。

前期，Echarts 库由百度开发，是国内图表库中的“领军人物”。目前，常见图表库框架有：Echarts、Highcharts 等。两个框架都需要使用浏览器的渲染技术，操作均比较简单，仍存在一些不同点<sup>[40]</sup>。

Highcharts 使用自由度更高，自定义选项更丰富，绘制数据图表具有更强的表现力，且不易发生延迟或卡顿<sup>[41]</sup>。相对而言，Echarts 在处理大量数据时，通

常需要借助如 dataZoom 等额外组件实现数据管理。Highcharts 是由国外开发，Echarts 由国内开发，因此 Echarts 说明文档比 Highcharts 更容易理解。Echarts 免费开源，Highcharts 用于商业用途时需要授权，个人使用含有水印，影响图表体验。

## 2.7 Shiro 安全机制

Shiro 是 Apache 下的一个 Java 安全框架，功能强大且易于使用。Shiro 框架从系统安全管理入手，为开发人员提供一套完整的系统安全解决方案，用于身份认证、权限授权、加密、会话管理等功能<sup>[42]</sup>。Shiro 框架具有完善的接口驱动设计，严格遵循面向对象的设计原则，支持各种自定义实现方案，表现出强大的扩展性<sup>[43]</sup>。

目前，常见的安全机制有：Shiro、SpringSecurity 等框架。两个框架都具有安全性和可靠性，仍存在着一些不同点。与 SpringSecurity 框架相比，Shiro 框架上手和实现更加容易，授权认证功能基本满足常规使用需求。Shiro 功能强大且简单灵活，支持独立运行，且不跟任何框架或容器绑定。

## 2.8 Redis 数据库

缓存中的数据存储在内存中，使用缓存进行数据库的查询，能极大加快数据检索速度和后台数据的响应速度，降低数据库的压力，提高用户的体验性<sup>[44]</sup>。

Redis 是一种基于内存存储原理的开源数据库，将数据存储在内存中，提供高速的读写操作，从而减轻应用程序数据库的负载<sup>[45]</sup>。与 SSD 或 HDD 存储的传统数据库相比，Redis 具有更高的性能和低延迟。Redis 作为 MySQL 数据库的缓存，帮助提高应用程序性能。通常将不经常更改且经常被请求的数据、任务关键性较低且经常变动的数据放入 Redis 数据库。

## 2.9 MyBatis 框架

MyBatis 是一个基于 Java 语言开发的持久层框架，它提供便捷的方式处理数

数据库中的数据。通过支持自定义 SQL 语句查询、高级映射以及存储过程的调用。MyBatis 对数据库操作进行封装，使用可扩展标记语言配置信息，将接口和 Java 对象映射成数据库中的数据，使得数据处理更加高效和方便<sup>[46]</sup>。

## 2.10 排序算法

数据结构中，排序算法是基础且关键的算法之一，主要包括交换排序（冒泡排序）、选择排序和插入排序。为提高排序效率，开发人员开发了上述基本算法的改进版本，包括快速排序、堆排序、希尔排序和 TopN 算法等。冒泡排序以简单直观而广为人知，快速排序因其高效的平均时间复杂度而被广泛应用于实际问题中。堆排序利用堆这一数据结构实现高效的排序，希尔排序则是通过比较分隔开的元素改进插入排序，提高对乱序数据的排序效率。TopN 算法是从许多的数值中选出前 N 个最大或最小的数值有序排列，最常见的应用于微博热搜榜、歌曲人气榜和投票选举等方面<sup>[47]</sup>。

## 2.11 权重算法

在分布式环境下，存在多个对等的服务节点时，为合理分配请求负载并考虑不同服务的性能差异，通常采用权重算法选出一个服务进行调用。计算权重的算法有：优序图法、AHP 层次法、熵值法、主成分法、因子分析法和权重乘积之和法等。

优序图法与 AHP 层次法均采用数字的相对大小信息计算权重。熵值法使用数据的熵值信息即信息量大小计算权重。主成分法与因子分析法是通过方差解释率计算权重。权重乘积之和算法是将权重赋值，再通过权重和系数相乘再相加得到一个新的值，最后再比较。

## 2.12 拟合算法

线性回归是一种统计学方法，利用一元线性回归方程对自变量和因变量之间的关系进行建模拟合，得到的模型是一个或多个参数的线性组合<sup>[48]</sup>。一般简单

回归模型中只有一个自变量，而多元回归模型则有多个自变量。拟合算法有：傅里叶级数拟合算法、Sin 和函数拟合算法和最小二乘法算法等。傅里叶级数拟合算法是一种强大而灵活的数学方法，能将复杂的函数拆解成多个简单的正弦和余弦函数的和。Sin 和函数拟合算法是一种基于正弦函数的拟合方法，用于逼近一组离散数据或连续函数。最小二乘法又称最小平方法，是回归分析中最基本的方法之一，用于拟合一个关于自变量和因变量的线性方程。通过最小二乘法算法进行线性回归模型的拟合，无限逼近模拟变量和实际变量，用最小化误差的平方和计算与数据最匹配的函数。

## 2.13 本章小结

本章主要概述了论文研究的相关理论知识，对整个系统使用的相关技术进行简要介绍。将 C/S 架构与 B/S 架构作比较，突出 B/S 架构的优势。对 MVC 模式进行了概述，了解其框架结构。阐述了 SpringBoot 框架和 Vue 框架的改进由来，分析了它们的优缺点。介绍了 Element-UI 框架，并和其他 UI 框架进行对比，突出了 Element-UI 框架的优点。分析了 Redis 数据库作为 MySQL 数据库缓存的优势，能够减少延迟和 MySQL 数据库的负载。最后介绍了排序算法、权重算法和拟合算法。通过整合这些框架技术，结合算法优化程序，实现更高效的运算和更快的响应速度，从而节省计算资源，提高系统的稳定性。



## 第三章 全能型供电所管理系统需求分析

### 3.1 需求分析概述

需求分析是整个系统开发过程中必不可少的重要一步,它是确保项目顺利进行和满足用户需求的关键所在。需求分析是指开发人员经过深入细致的调查、研究并分析,准确理解用户和系统的功能、性能、可靠性等具体要求,消除开发人员与用户之间的认知差异,最终形成一个完整、清晰、一致的需求说明。

近年来,全球正经历一场数字化转型的浪潮,以大数据、云计算和人工智能等为代表的数字技术加速发展,广泛应用融入经济社会发展各领域<sup>[49]</sup>,供电企业作为国民经济发展的重要支柱,更应推进数字化与生产经营管理深度融合。目前本人在乡镇供电所工作,接触到各管理专业的工作内容,各专业流转至供电所层面的业务应用繁多,缺少统一的管理手段和辅助工作改进的系统工具,这大大降低了供电所的管理水平和工作效率。

基于供电所面临的现状,对各专业的数据融合分析,横向业务贯穿,完成数字化转型,真正实现基层减负,大幅度减轻人工管理工作难度和工作量,提高人员工作效率。

### 3.2 用户需求分析

该系统针对供电所及上级公司不同用户的需求进行相应的功能权限管理,系统用户主要有系统管理员、公司管理员、供电所管理人员与普通员工。下面对这四类用户的需求进行介绍:

#### (1) 系统管理员

系统管理员拥有本系统的全部管理权限,负责系统日常运营维护、系统升级以及更新系统数据信息等职能。还需要对接上级公司开展调研工作,定期向供电所管理人员汇报分析,为管理人员的科学决策提供强有力的数据支撑。

#### (2) 公司管理员

公司管理员具有管理辖区所有供电所的权限,主要分为三点:一是修改部门

权限和用户信息。发生供电所合并或者人员调动时，公司管理员在系统管理模块修改部门信息和用户信息。二是分配菜单功能的使用权限。将系统用户按角色划分使用权限，确保不同用户只能访问其所需的功能和数据。三是管理系统里全部客户的档案信息。

### （3）供电所管理人员

供电所管理人员分为所长、核算员、营销员和安技员。供电所的日常管理工作内容划分为四大专业，分别是：综合管理类、核算类、营销类和安全生产类。管理人员与四大专业相对应，所长负责综合管理类专业以及供电所整体事务统筹安排工作。综合管理类是所务公开与员工管理、核算类主要是负责客户电费的发行与回收、营销类分成台区线损计量与营业业务、安全生产类为配电设备的巡视检查以及抢报修。将不同类别的工作内容划分为四大专业再分配到供电所各管理人员的角色，做到分工明确，相互配合，统筹兼顾。

### （4）普通员工

供电所普通员工为运维采集工，运维采集工又称台区经理，主要负责所辖台区的配网线路运维、抢报修、客户电费回收和计量采集维护等工作。

## 3.3 功能性需求分析

功能性需求是系统或产品应具备的具体功能或行为，它描述了软件应如何运行才能实现预期目标。本系统的主要功能是通过整合各管理专业工作要求，简化工作流程，提供给供电所管理人员使用。系统的需求分析围绕供电所的总体目标展开，目的是对供电所各项工作的运行实现高效管理，减少不必要的人力、物力资源投入，实现供电所管理信息化。按照功能性需求分类，将全能型供电管理系统分为：工作看板、指标看板、营销服务、核算管理、工单管理、安全生产、综合管理和系统管理等 8 个模块。本文采用 UML 用例图的方式，用于描述系统各个功能模块的需求。系统总体功能用例图如图 3-1 所示：

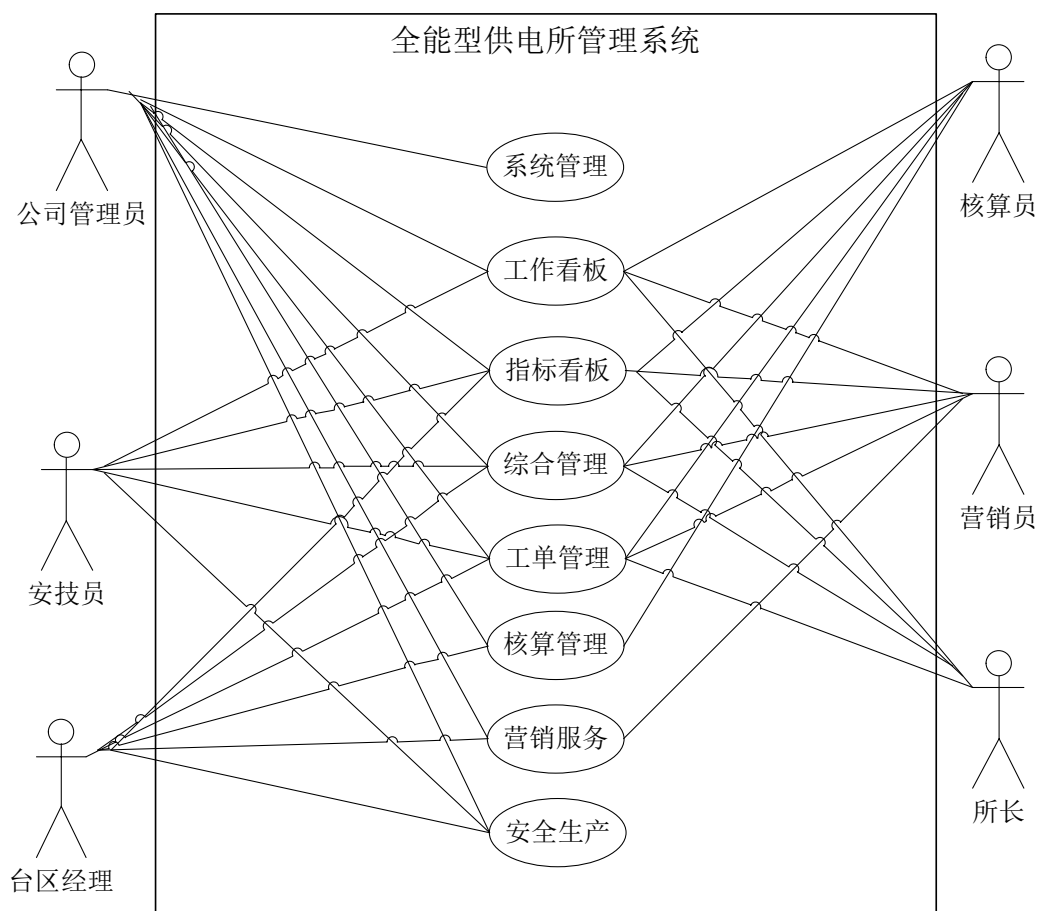


图 3-1 系统总体功能用例图

### 3.3.1 系统管理模块

系统管理模块需要包含系统用户管理、部门管理、菜单管理、角色管理、客户档案信息管理和日志管理。由于供电所数量众多，队伍庞大，目前又面临合并和人员调动现象，需要维护用户信息及部门信息，并设置相应供电所管理员，有利于维护所在供电所的相关数据信息。同时，对不同用户的需求进行相应的功能权限管理，设置角色管理，加以权限控制，更好地管理各专业工作内容。系统管理模块用例图如图 3-2 所示：

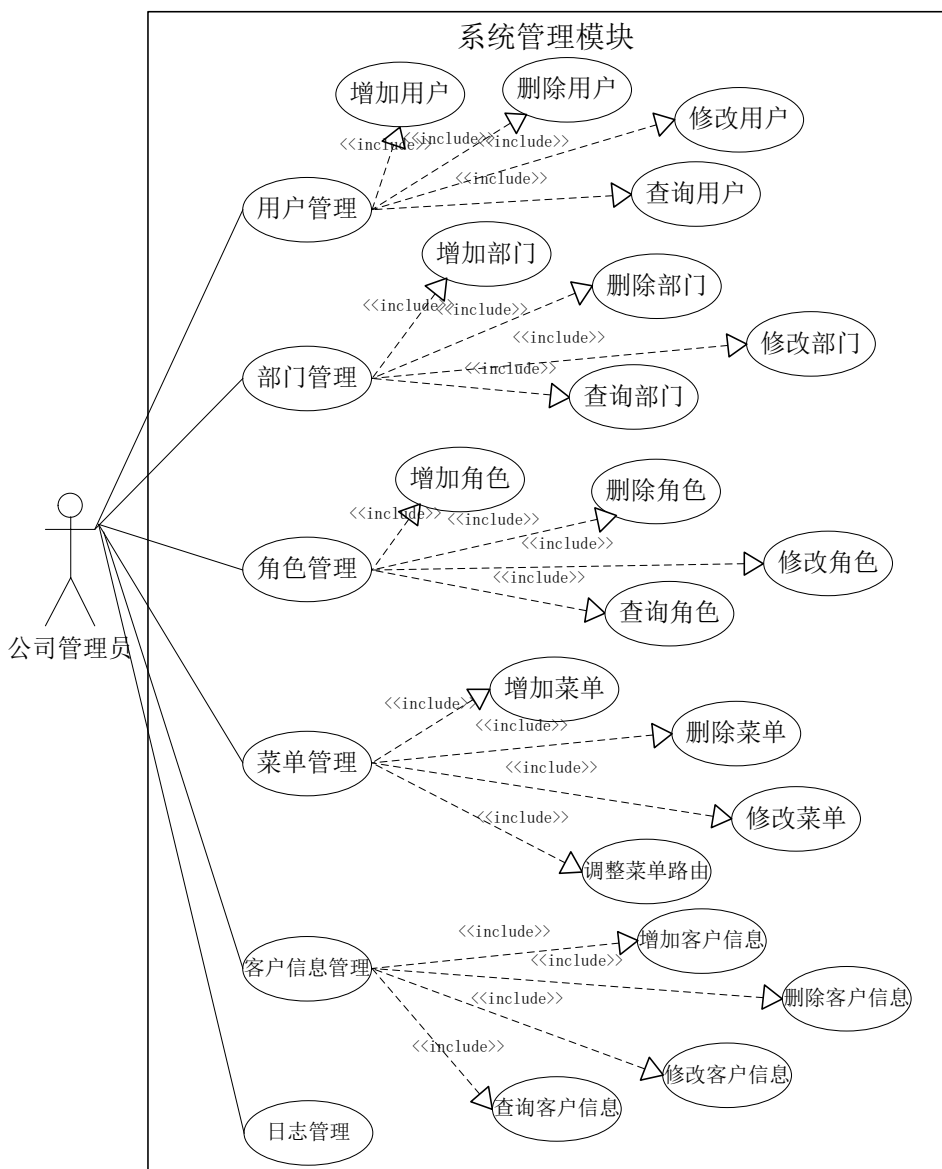


图 3-2 系统管理模块用例图

### 3.3.2 工作看板模块

为方便各管理专业日常工作所需指标数据的进行统计与展示，将工作看板分配到各管理专业，使管理人员更加高效地进行工作规划和任务分配。工作看板模块用例图如图 3-3 所示：

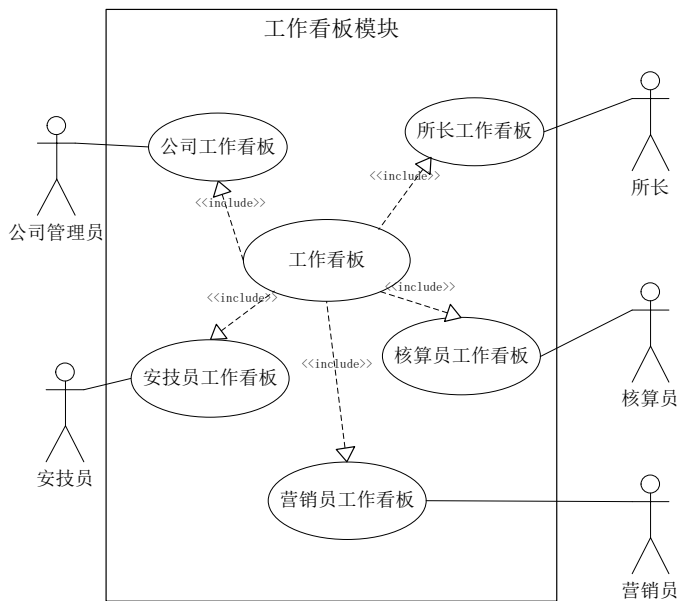


图 3-3 工作看板模块用例图

### 3.3.3 指标看板模块

指标看板模块需要展示各供电所同业对标水平，以年月日为单位对比各个供电所的业务指标，涵盖各管理专业重要指标数据，同时以排名的形式展示，不断促进供电所业务水平提升，激发员工争先创优的精神。指标看板模块又分成生产运维、营销服务和综合管理三个子模块。指标看板模块用例图如图 3-4 所示：

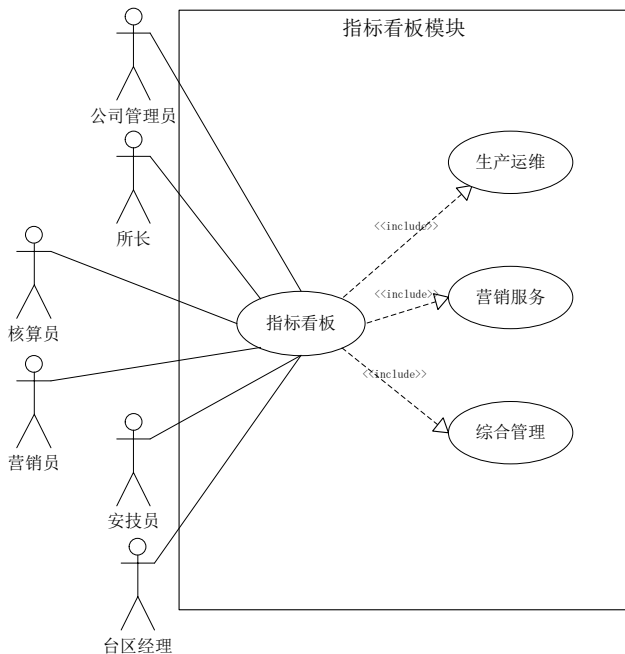


图 3-4 指标看板模块用例图

### 3.3.4 营销服务模块

输电线路传输电能会产生能量损耗，称为线路损耗，简称线损。线损率是线损电量占供电量的百分比，是衡量线损高低的标志，是营销服务的一个关键指标。同时也是一项重要的经济技术指标，反映了电网的规划设计水平、生产运行水平和经营管理水平。

线损按台区划分，每个台区都有其台区线损率，因客户的每日用电量不同，导致每日的台区线损率处于浮动状态。当遇到客户窃电、电表烧坏、客户超容和线路末梢大电量等原因，会出现线损率异常。在营销服务中，需要详细展示供电所台区线损数据，并按线损类别分类汇总，方便营销员查看线损异常台区，及时通知台区经理前去现场核查线损异常原因。营销服务模块用例图如图 3-5 所示：

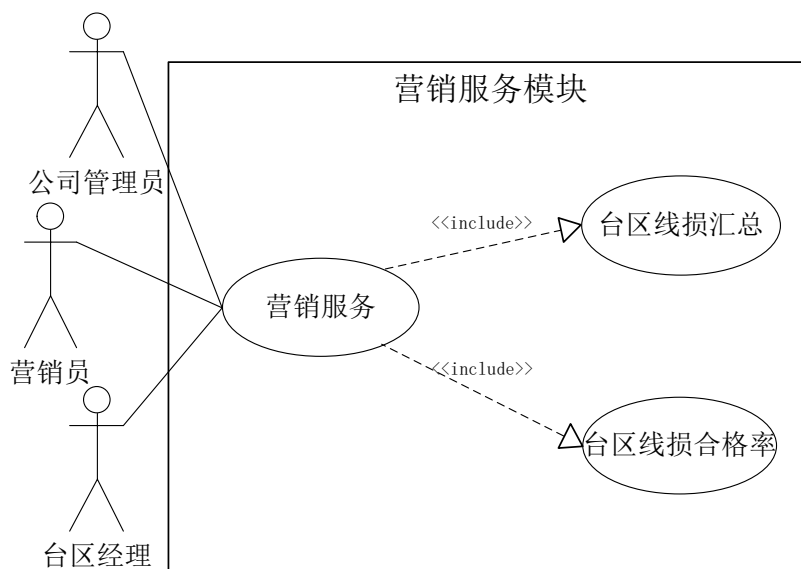


图 3-5 营销服务模块用例图

### 3.3.5 核算管理模块

电费是电网公司经营的成本，是最大经济收入来源，直接关系到电网公司的效益<sup>[50]</sup>。今年 4 月，国网江苏省电力公司发布关于《关于实施按自然月抄表结算的公告》，于 6 月 1 日起，全省居民客户和低压非居民客户调整为每月抄表，按月结算，可见电费回收的重要性。在核算管理模块中需要展示供电所所辖范围

内的客户电费缴费情况，由台区经理负责完成所辖区域的电费回收。当客户出现欠费情况时，核算员及时将欠费客户生成工单，派发给台区经理进行处理，提高电费回收率。同时，电费明细支持导出并打印，台区经理查看客户信息并催缴。核算管理模块用例图如图 3-6 所示：

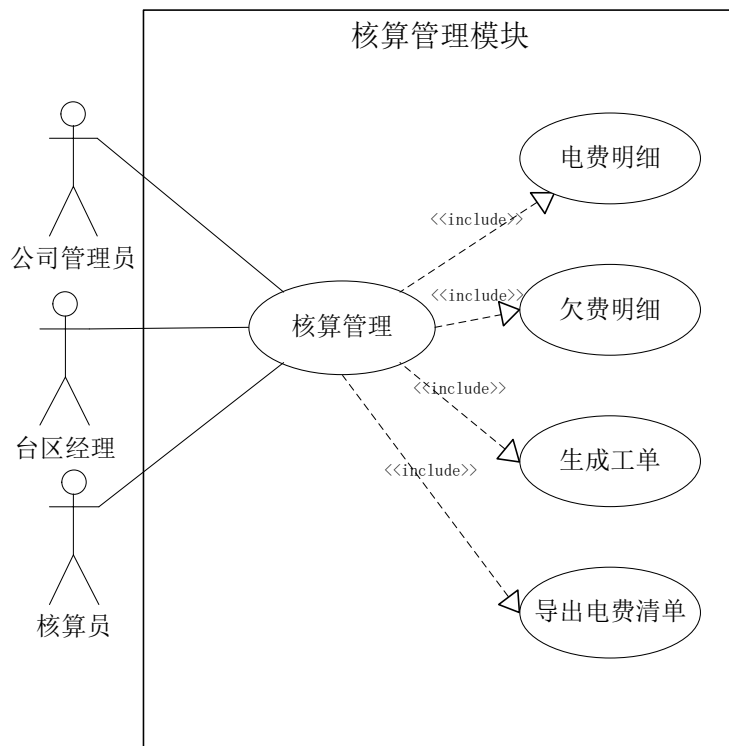


图 3-6 核算管理模块用例图

### 3.3.6 工单管理模块

工单管理模块是本系统中的一个重要模块，它用于管理供电所内部的工单流程。工单是指台区经理的各种工作任务，通过工单管理模块对所有工单进行统一的管理和跟踪，确保工单的快速处理和及时跟进，实现台区经理的工作全过程监督、工作结果实时反馈、工作内容实时归档等功能，避免了工单的遗漏和延误，大大降低因执行过程中因疏忽而导致的安全风险，同时提高工作过程透明化和标准化，实现工作效率和服务质量双提升。工单管理模块用例图如图 3-7 所示：

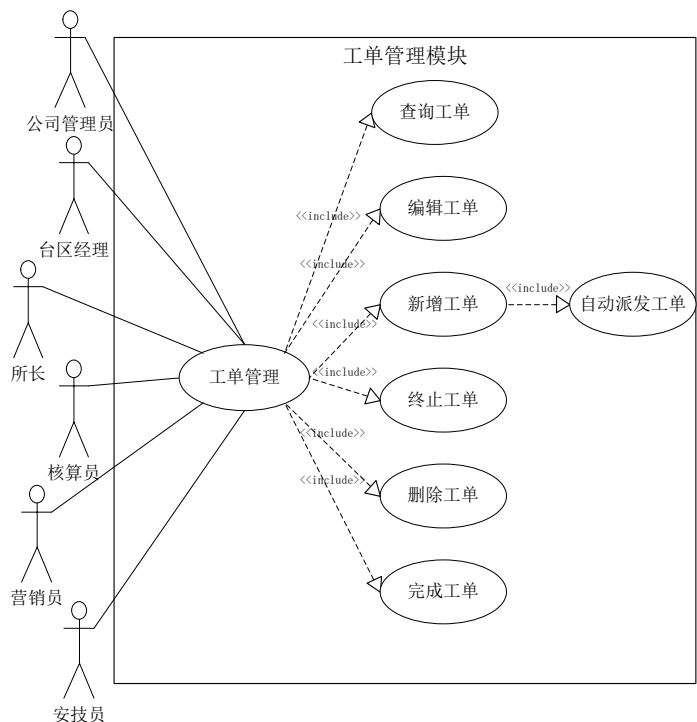


图 3-7 工单管理模块用例图

### 3.3.7 安全生产模块

大部分的乡镇供电所都地处经济欠发达地区，受此影响，乡镇供电所的资金投入和设备投入有所欠缺。导致乡镇配网线路跳闸率和计量装置受损率高于城区，客户故障报修数量增加。在安全生产模块需要详细记录着客户的故障报修信息，及时生成工单并通知台区经理前去客户指定地点处理报修故障。安全生产模块用例图如图 3-8 所示：

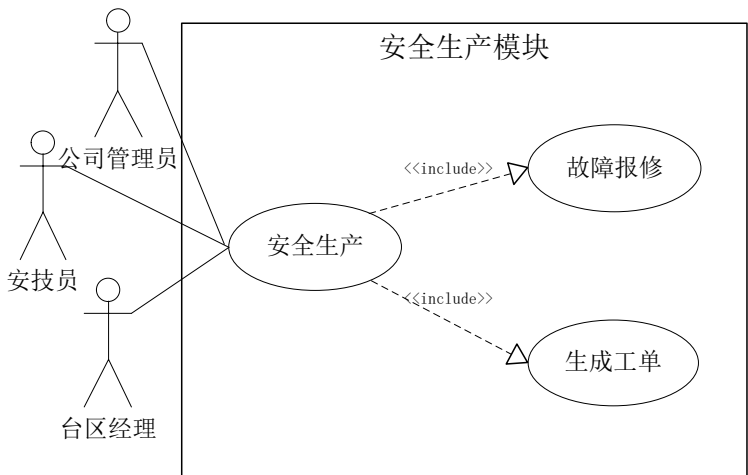


图 3-8 安全生产模块用例图



### 3.3.8 综合管理模块

综合管理模块需要展示供电所所务公开信息和员工信息，确保上级公司领导顺利进行调研和考察。供电所所长负责维护供电所基础信息和员工基本信息，比如供电所管辖台区数量、客户数量、管辖乡镇数量等信息。综合管理模块用例图如图 3-9 所示：

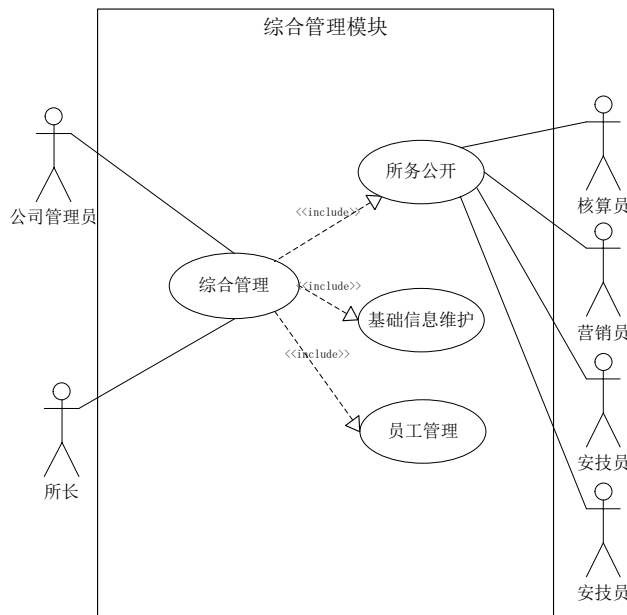


图 3-9 综合管理模块用例图

### 3.4 非功能性需求分析

系统非功能性需求是指除功能需求以外，系统满足用户业务需求而必须具有的其他特性，比如系统性能、安全性、兼容性、易用性等。因此，还需确保该系统满足以下几方面要求：

#### (1) 性能：

在不同环境和压力下系统可以正常运转。系统的平均响应时间应在 3 秒以内，系统能够至少支持 200 个用户的同时访问压力，服务器每秒处理的请求事务数至少为 40 个。

#### (2) 安全性：

系统标准化控制用户角色功能：

①系统对用户身份进行统一设置,系统的每个用户都将拥有唯一的系统用户账号<sup>[51]</sup>;

②系统中的每个用户账号均对应一个角色;

③将每个角色分配对应的系统模块使用功能,角色只能使用已分配使用权限的功能模块。

(3) 兼容性:

做好浏览器兼容,确保系统在常用的浏览器下都能正常显示。

(4) 易用性:

系统需要简单且易用,使用户可以在 10 分钟内轻松掌握。系统易用性要求系统界面简洁和直观,系统功能需达到简单易用的要求。

(5) 维护性:

对于用户的变动,系统是可维护的,能够通过增加或删除用户权限完成,对于系统界面以及数据字段都具有可维护性。

(6) 扩展性:

系统代码、业务环境等方面应具备良好的可扩展性,确保系统的更新和升级,以适应不断变化的工作要求。

(7) 实用性:

在保证系统正常运行的前提下,设计与实际工作内容相适应的系统,提高工作效率,简化工作复杂度。

(8) 可靠性:

系统应采取多方面措施,以确保系统的高可靠性及高稳定性,保证系统的正常运行。

### 3.5 可行性分析

对系统需求进行分析后,通过深入的可行性研究,才能确定本系统开发的可行性,从而做出明智的决策。系统可行性分析是从多个层面判断该系统在当前技术条件下是否实现,在有限的开发的成本下是否开发完整,在当下电脑环境下是

否可以运行。下面从技术、经济、系统运行三个方面研究本系统的可行性。

(1) 技术可行性:

该系统采用前后端分离技术,使得前端和后端独立开发和演化,提高开发效率、可维护性和可扩展性。经过反复技术对比和选型,确定了 Vue 为前端技术框架,结合 ElementUI 框架和 ECharts 库,提供可视化数据展示。后端使用 SpringBoot 框架简化系统开发所需要的复杂配置,将 Redis 和 MySQL 数据库结合使用,提高数据访问效率。这些技术完全能够满足全能型供电所管理系统的开发需求。

(2) 经济可行性:

该系统基于成熟的软硬件环境,系统开发成本不再需要额外的昂贵费用,后期只需要定期维护系统即可,在经济上完全可行。

(3) 系统运行可行性:

该系统基于 B/S 架构开发,只需要一台装有浏览器的电脑就能运行。用户操作几乎全部在浏览器上进行,用户的请求、处理等操作都在服务端进行,不占用用户电脑的计算资源,对用户电脑的硬件要求较低,在系统运行上完全可行。

### 3.6 本章小结

本章首先概述了需求分析的重要性,对用户需求进行详细分析。其次基于系统需求设计 8 个功能模块,对每个模块进行分析,明确其具体需求和具体功能,并绘制了 UML 用例图。最后通过从技术、经济、系统运行三个方面研究本系统的可行性。

## 第四章 全能型供电所管理系统详细设计

本章在第三章全能型供电所管理系统需求分析的基础上,对系统进行详细设计,包括系统总体架构设计、功能模块设计、数据库设计和算法设计。本章中的设计充分考虑了软件的易理解性、可拓展性、可重用性与可维护性,坚持面向对象的设计原则,详细地描述了整个系统的设计过程<sup>[52]</sup>。

### 4.1 系统总体架构设计

全能型供电所管理系统主要由工作看板、指标看板、营销服务、核算管理、工单管理、安全生产、综合管理、系统管理和登录注册模块组成。当用户登录时,通过传输层 HTTP 请求和响应,使用 Shiro 对权限安全进行验证,严格控制权限访问及系统安全。应用层为不同功能提供相应的应用服务,数据层使用 MySQL 数据库存放业务数据和操作日志数据,将不经常更改且经常被请求的数据、任务关键性较低且经常变动的数据放入 Redis 数据库,提高系统的响应速度。系统架构图如图 4-1 所示:

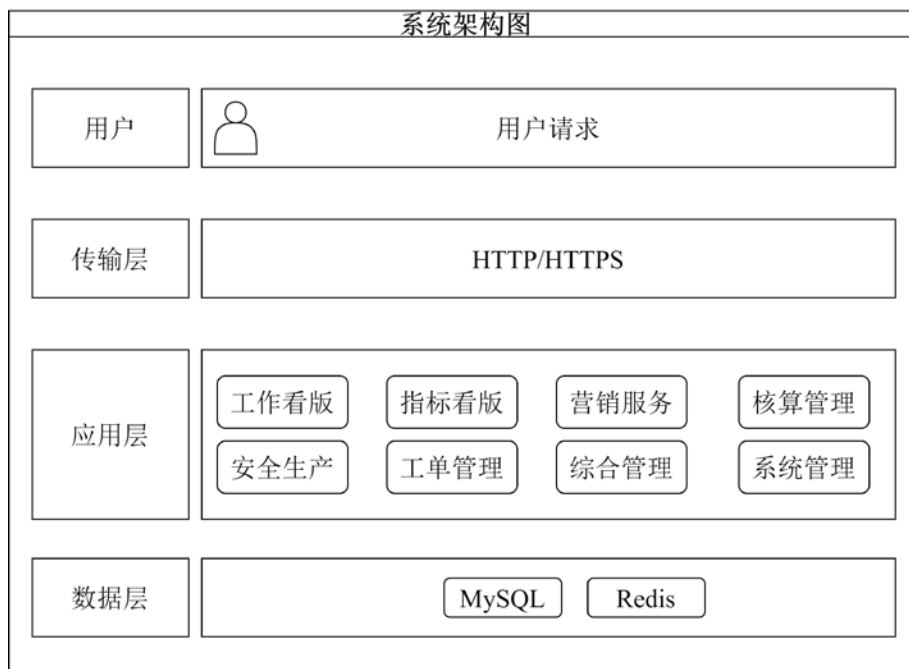


图 4-1 系统架构图

本系统采用 B/S 架构和 MVC 设计模式，基于 SpringBoot 框架和 Vue 框架，实现了前后端分离。前端采用的技术有 Vue 框架、Element-UI 框架、Echarts 库、Mybatis 框架，后端采用的技术有 SpringBoot 框架、TK-Mapper 框架、Shiro 安全机制、Aop 技术、lombok 库、TopN 算法、权重算法和最小二乘法算法等，本文第二章已详细介绍这些技术算法。系统技术架构图如图 4-2 所示：

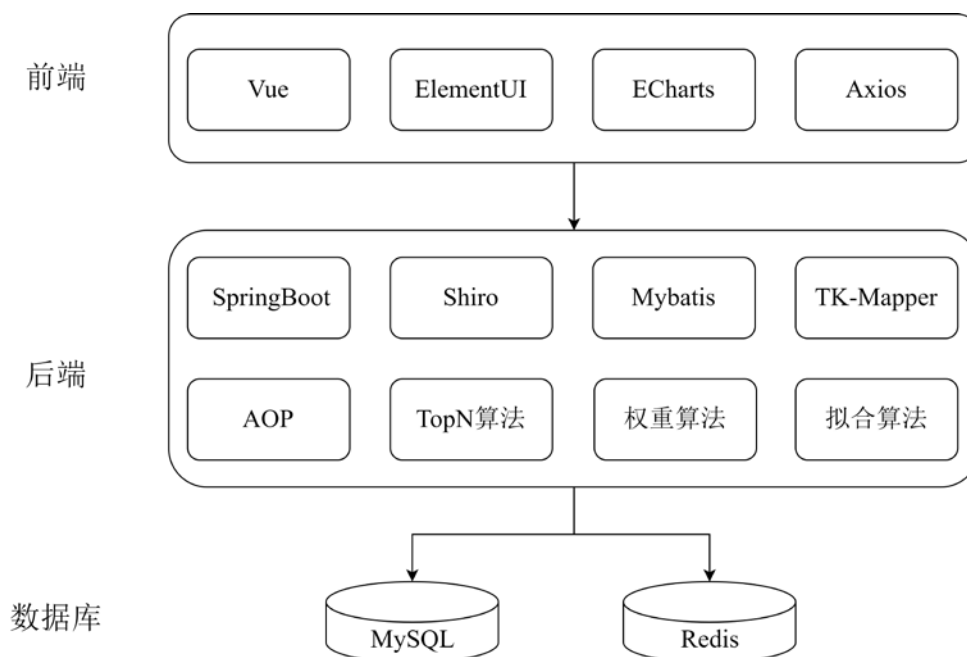


图 4-2 系统技术架构图

## 4.2 系统功能模块设计

本文研究的全能型供电所管理系统主要包括 8 个功能模块，分别是：工作看板模块、指标看板模块、营销服务模块、核算管理模块、工单管理模块、安全生产模块、综合管理模块和系统管理模块。基于本人在供电所工作中遇到的一些问题，因多个系统间的数据没有实现共享，系统频繁切换、操作步骤繁琐，严重降低了基层员工工作效率和管理水平。该系统将着力解决这些问题，从根本上提升供电所的管理质效。系统功能模块图如图 4-3 所示：

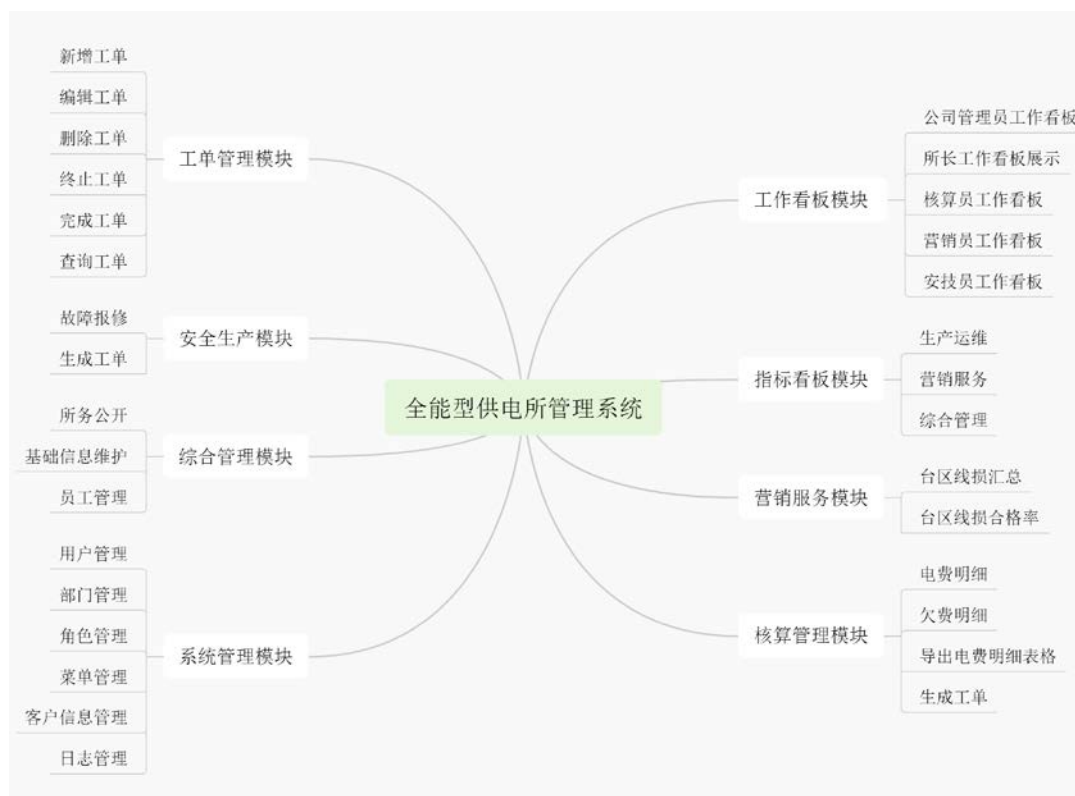


图 4-3 系统功能模块图

#### 4.2.1 系统管理模块

系统管理模块主要划分为用户管理、部门管理、角色管理、菜单管理、客户信息管理和日志管理这六个功能子模块。用户管理存放系统中的所有用户信息。参照公司的组织架构，部门管理用于统一维护管理系统中的部门信息。角色管理用于区分系统不同的使用权限。菜单管理支持用户自定义个性化的菜单功能，系统界面的菜单实现导航跳转。客户信息管理存放全部客户档案信息。日志管理记录系统中的所有用户的操作记录。

在该模块中，公司管理员具有以上六个功能子模块的操作权限。系统管理的主要操作流程为：公司管理员进入系统后，选择系统管理功能模块，支持用户、部门、角色等信息的查询、删除、添加和修改。允许用户登录或者禁用系统，角色依据权重排序，为角色分配菜单权限，调整系统菜单界面。随时查看系统操作日志，确保系统的安全性。

#### 4.2.2 工作看板模块

工作看板模块包括公司管理员工作看板、所长工作看板、核算员工作看板、营销员工作看板和安技员工作看板。管理员工作看板为公司管理层提供整个市范围内的供电所三大重要指标（线损合格率、采集成功率和电费回收率）及所辖区域分布图，帮助公司管理员全面了解供电所的地域分布和三大重要指标情况。所长工作看板侧重于电费回收率、采集成功率等关键数据。其中，电费回收率通过柱状图和折线图的组合展示，使所长能够直观地了解各台区经理负责的电费回收情况，便于及时发现并解决问题。核算员工作看板主要显示供电所电费回收情况、逾期缴费统计等数据。营销员工作看板主要关注采集成功率、线损率等营销指标。安技员工作看板记录着生产运维基础数据。通过数据的及时更新和展示，使得安技员能够及时发现潜在的安全隐患，确保电网设备安全可靠的运行。

在这个模块中，系统会根据登录用户的身份自动判断其所属的角色，并基于该角色的权限和配置，动态展示相应的工作看板。工作看板模块操作流程如图 4-4 所示：

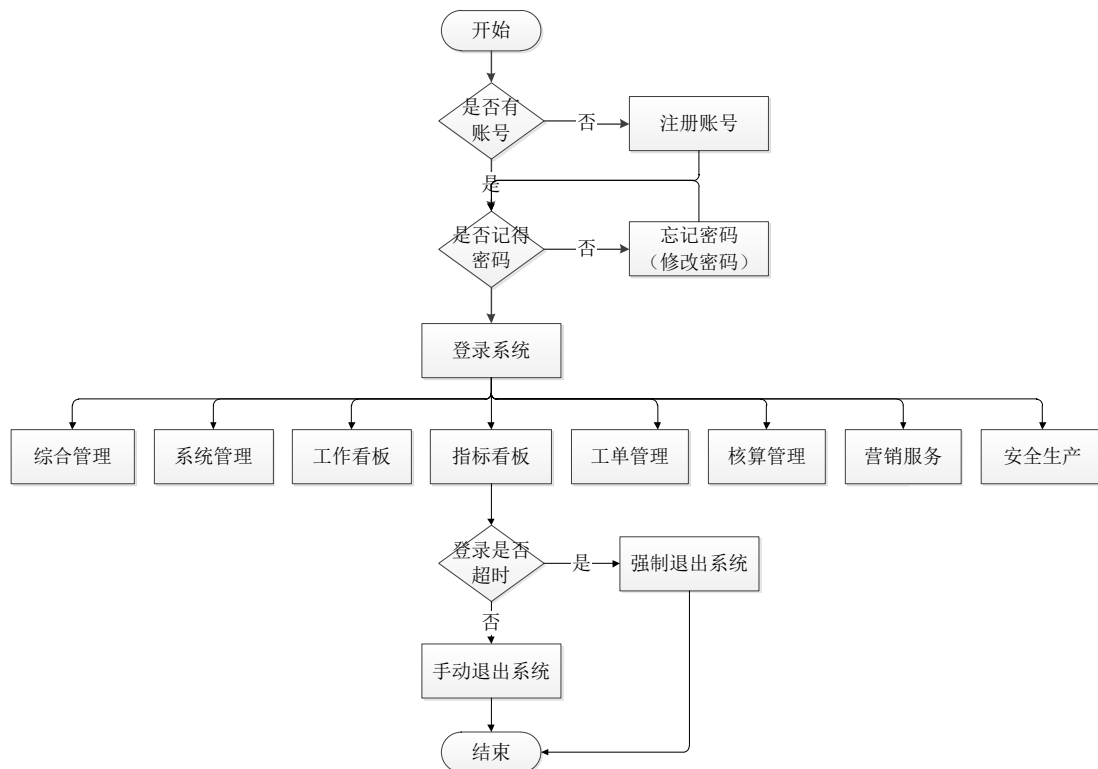


图 4-4 工作看板模块操作流程

#### 4.2.3 指标看板模块

指标看板模块划分为生产运维、营销服务和综合管理这三个功能子模块。三类指标采用全县平均值、全县最优值及在全市排名值的形式同业对标，形成比超赶学的良好竞争氛围。

#### 4.2.4 营销服务模块

营销服务模块包括台区线损汇总和台区线损合格率这两个功能子模块。台区线损汇总可以更清晰地展现台区经理管辖台区线损情况。通过营销员查看台区线损合格率，第一时间发现异常台区，锁定异常设备，减少异常电量损失。

在该模块中，营销员重点关注台区经理管辖的台区线损类别和线损合格率。同时查看线损明细，发现异常台区，及时通知台区经理现场勘察并处理异常。

#### 4.2.5 核算管理模块

核算管理模块主要分为电费明细、欠费明细、导出电费明细表格和生成工单这四个功能子模块。电费明细记录供电所辖区客户电费缴费明细，电费催缴工作按台区划分，台区经理负责完成。在该模块中，核算员进入电费明细模块查看台区经理管辖台区客户缴费信息，选择欠费明细模块，生成欠费工单派发给相应的台区经理，督促台区经理现场完成电费回收工作。电费明细支持导出并打印，易于台区经理查看客户信息。核算管理模块的操作流程泳道图如图 4-5 所示：

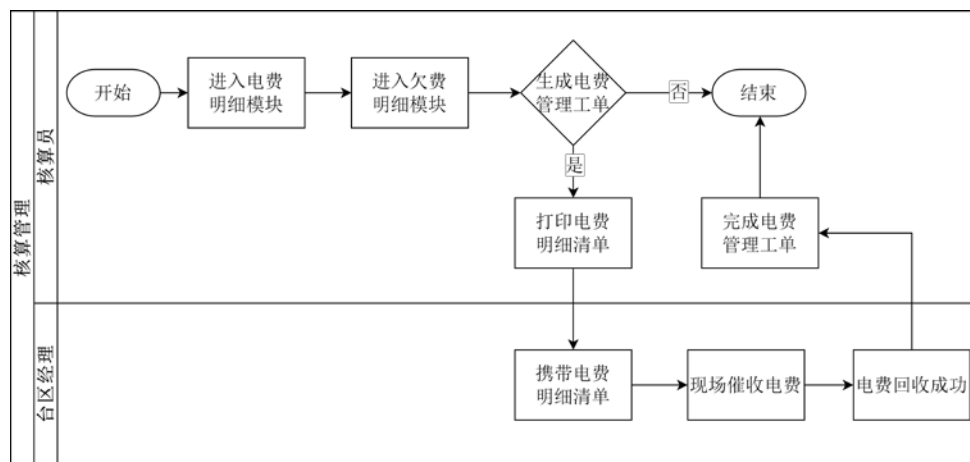


图 4-5 核算管理模块的操作流程泳道图



#### 4.2.6 工单管理模块

工单管理模块包含新增工单、编辑工单、删除工单、终止工单、完成工单和查询工单这六个功能子模块。新增工单主要是将工作以工单的形式派发给台区经理完成。编辑工单用于修改工单内容。删除工单可以删除作废的工单记录。终止工单是强行终结未完成的工单。完成工单是标记全流程无误的工单。查询工单用于查询所有工单状态及工单信息。

在该模块中，管理人员选择与实际工作一致的工单分类，填写工单内容，生成工单，派发给台区经理处理。未在规定时间内完成成为超时工单，因特殊情况无法在规定时间内完成则需要终止工单。同时支持“执行单位”、“工单编号”、“工单责任人”、“派工人”、“派单日期”、“工单分类”、“工单状态”等内容进行筛选查询指定工单，缩小工单查询范围。工单管理模块操作流程时序图如图4-6所示：

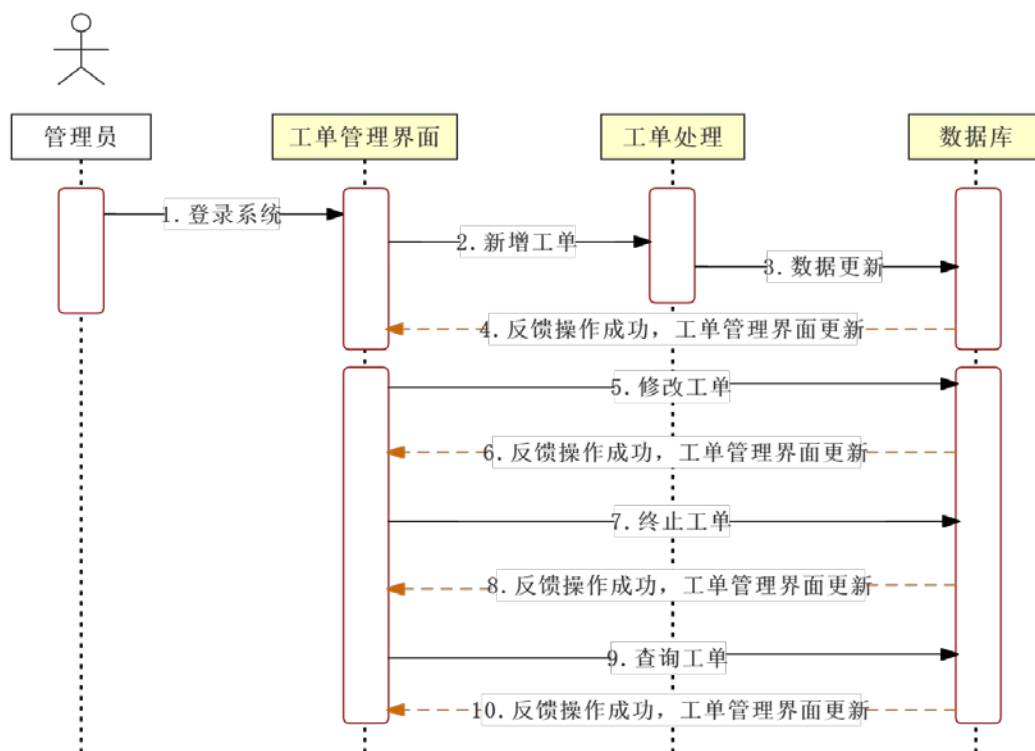


图 4-6 工单管理模块操作流程时序图

#### 4.2.7 安全生产模块

安全生产模块主要划分为故障报修和生成工单两个功能子模块，故障报修模块标记辖区客户抢报修记录。生成的工单支持与故障报修关联，监督台区经理现场完成抢报修工作。

在该模块中，安技员根据“报修户号”查询，添加“报修内容”，同时生成工单派发给相应的台区经理去处理。实现用“客户编号”、“报修时间”查询用户历史报修内容。安全生产模块的操作流程泳道图如图 4-7 所示：

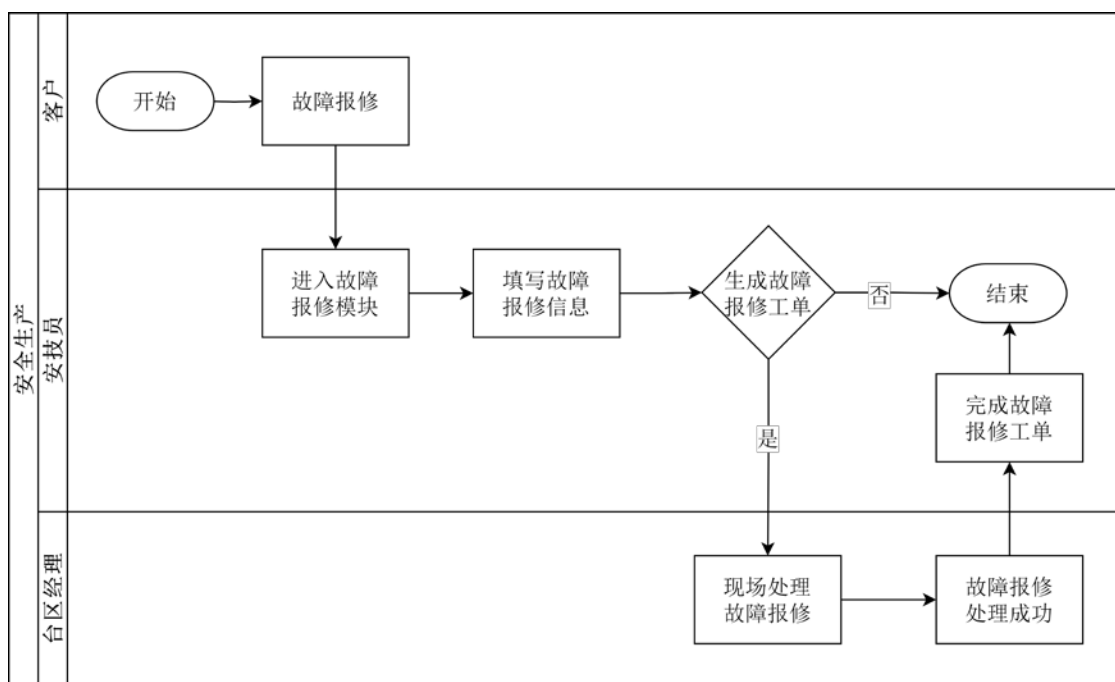


图 4-7 安全生产模块的操作流程泳道图

#### 4.2.8 综合管理模块

综合管理模块主要包含所务公开、基础信息维护和员工管理这三个子模块。通过基础信息维护修改所务公开信息，并对外展示供电所的基本信息。员工管理存放着所有的员工信息。所长和公司管理员能够维护基础信息及员工信息。综合管理模块操作流程时序图如图 4-8 所示：

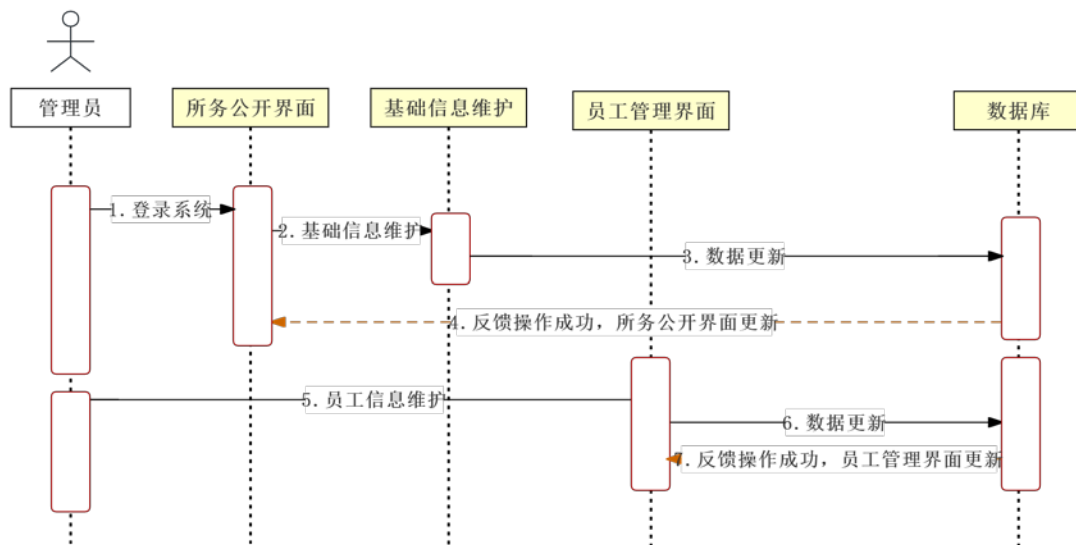


图 4-8 综合管理模块操作流程时序图

## 4.3 数据库设计

### 4.3.1 数据库概念设计

全能型供电所管理系统采用了 MySQL 作为数据库, 根据系统各个模块的功能设计出与之匹配的数据表和系统数据库模型图, 更加直观地看出表与表之间的关联。

通过更清晰反映现实世界里事物之间的关联, 根据本系统的数据库结构, 将系统需求转化为数据库中实体、属性以及两者之间关系, 设计出数据模型结构。基于本系统的功能模块以及系统需求, 涉及用户、员工、工单、菜单、报修、电费缴纳、台区经理、台区、台区线损、缴费渠道、电费回收等实体。按照实体之间的关联, 分别从用户和公司管理员角度, 设计出系统 E-R 图如图 4-9 和 4-10 所示:

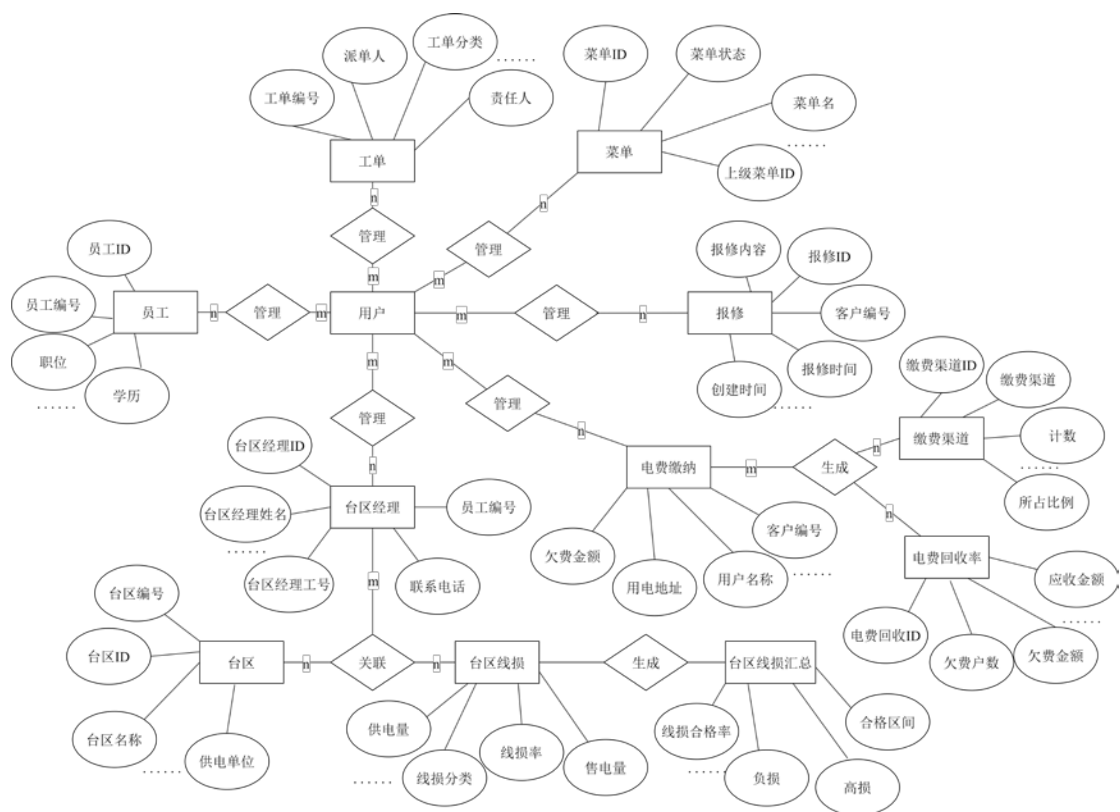


图 4-9 用户角度 E-R

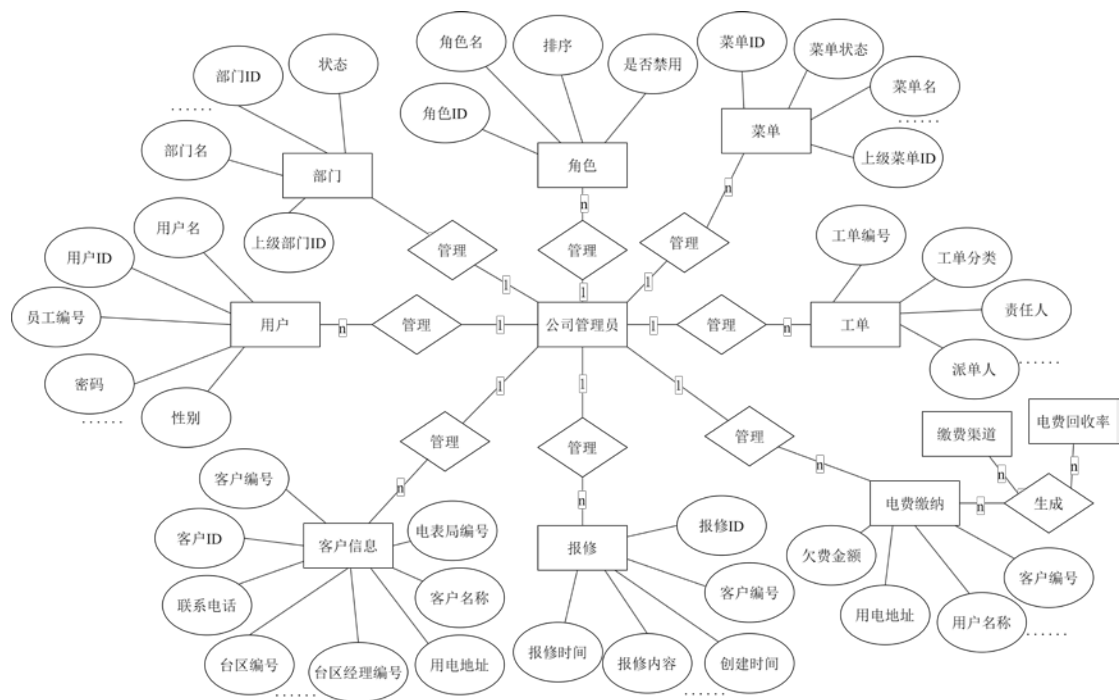


图 4-10 公司管理员角度 E-R 图

### 4.3.2 数据库表设计

根据 E-R 模型可知, 本系统主要包括用户信息表、用户与角色关联表、角色表、菜单角色关联表、菜单表、用户部门关联表、部门表、员工信息表、台区信息表、台区线损表、台区线损汇总表等信息表, 选取营销服务模块、核算管理模块和安全生产模块的重要数据表信息如下:

#### 4.3.2.1 营销服务模块

该模块主要是对供电所台区信息以及台区线损变动等进行管理, 设计的数据表主要有台区经理、台区、台区线损的信息表, 下面是营销服务模块主要数据表的设计详情。

##### (1) 台区信息表

台区信息表中存放着所辖台区的所有信息, 将所辖客户划分到各个台区, 再关联台区经理, 有利于台区管理, 包含台区编号、台区名称、台区经理姓名、台区经理工号、台区经理联系电话等字段, 其中台区编号是主键, 具体的表结构如表 4-1 所示:

表 4-1 台区信息表

编号	字段名	数据类型	非空	主键	注释
1	lob_no	varchar(20)	是	是	台区编号
2	lob_title	varchar(200)	是		台区名称
3	lob_manager_name	varchar(200)	是		台区经理姓名
4	lob_manager_no	varchar(20)	是		台区经理工号
5	phone	varchar(20)	是		台区经理联系电话
6	power_unit	varchar(200)	是		供电单位

##### (2) 台区线损表

台区线损表中记录着所辖台区线损的所有信息, 详细分类涉及台区线损的参数, 包含台区线损编号、台区编号、台区名称、台区经理姓名、台区经理工号、供电量、售电量、线损率、线损量等字段, 其中台区线损编号是主键, 具体的表结构如表 4-2 所示:

表 4-2 台区线损表

编号	字段名	数据类型	非空	主键	注释
1	id	bigint(19)	是	是	台区线损编号
2	lob_no	varchar(20)	是		台区编号
3	lob_title	varchar(200)	是		台区名称
4	lob_manager_name	varchar(200)	是		台区经理名称
5	lob_manager_no	varchar(20)	是		台区经理工号
6	power_supply	varchar(20)	是		供电量
7	power_sale	varchar(20)	是		售电量
8	power_loss	varchar(20)	是		线损量
9	loss_rate	varchar(20)	是		线损率
10	loss_type	varchar(20)	是		线损类别
11	power_unit	varchar(200)	是		供电单位

## (3) 台区线损汇总表

台区线损汇总表用于统计所辖台区线损的汇总数据，与台区线损表相关联，汇总数据能直观看到台区线损总体情况，在第一时间通知相关责任人前去现场处理，包含台区线损汇总编号、台区编号、台区经理工号、运行台数、负损、正常、高损、合格区间、线损合格率等字段，其中台区线损汇总编号是主键，具体的表结构如表 4-3 所示：

表 4-3 台区线损汇总表

编号	字段名	数据类型	非空	主键	注释
1	id	bigint(19)	是	是	台区线损汇总编号
2	lob_manager_name	varchar(200)	是		台区经理名称
3	lob_no	varchar(20)	是		台区编号
4	lob_manager_no	varchar(20)	是		台区经理工号
5	run_no	int(11)	是		运行台数
6	loss_no	int(11)	是		负损
7	normal_no	int(11)	是		正常
8	h_loss_no	int(11)	是		高损
9	quali_range	int(11)	是		合格区间
10	unquali_range	int(11)	是		不合格区间
11	loss_quali_rate	varchar(20)	是		线损合格率
12	power_unit	varchar(200)	是		供电单位
13	c_time	datetime	是		创建时间

#### 4.3.2.2 核算管理模块

该模块主要是对供电所辖区客户、电费缴纳、缴费渠道等信息以及电费回收变动等进行管理，设计的数据表主要有客户、电费缴纳、缴费渠道、电费回收率的信息表，下面是核算管理模块主要数据表的设计详情。

##### (1) 客户信息表

客户信息表保存着所辖台区客户的详细信息，包含客户编号、电表局编号、客户名称、用电地址、联系电话、台区编号、台区名称、台区经理工号、台区经理名称、供电单位等字段，其中客户编号是主键，具体的表结构如表 4-4 所示：

表 4-4 客户信息表

编号	字段名	数据类型	非空	主键	注释
1	oprs_no	bigint(19)	是	是	客户编号
2	power_no	bigint(19)	是		电表局编号
3	oprs_name	varchar(20)	是		客户名称
4	oprs_address	varchar(20)	是		用电地址
5	oprs_number	varchar(200)	是		联系电话
6	lob_no	varchar(20)	是		台区编号
7	lob_title	varchar(200)	是		台区名称
8	lob_manager_no	varchar(20)	是		台区经理工号
9	lob_manager_name	varchar(20)	是		台区经理名称
10	power_unit	varchar(200)	是		供电单位

##### (2) 电费缴纳表

电费缴纳表统计着所辖台区客户的电费缴纳信息，包含电费缴纳编号、客户编号、客户名称、用电地址、联系电话、台区编号、台区名称、台区经理工号、台区经理名称、应收电量、应收金额、实收金额等字段，其中电费缴纳编号是主键，具体的表结构如表 4-5 所示：

表 4-5 电费缴纳表

编号	字段名	数据类型	非空	主键	注释
1	id	bigint(19)	是	是	电费缴纳编号
2	oprs_no	varchar(20)	是		客户编号
3	oprs_name	varchar(20)	是		客户名称
4	power_address	varchar(200)	是		用电地址
5	phone	varchar(20)	是		联系电话
6	lob_no	varchar(20)	是		台区编号
7	lob_title	varchar(200)	是		台区名称
8	lob_manager_name	varchar(200)	是		台区经理名称
9	lob_manager_no	varchar(20)	是		台区经理工号
10	power_unit	varchar(200)	是		供电单位
11	able_power	varchar(20)	是		应收电量
12	able_amt	varchar(20)	是		应收金额
13	paid_amt	varchar(20)	是		实收金额
14	overdue_amt	varchar(20)	是		欠费金额
15	pay_chl	varchar(20)	是		缴费渠道
16	c_time	datetime	是		创建时间
17	u_time	datetime	是		更新时间

### (3) 电费回收率表

电费回收率表记录着所辖台区客户的电费回收情况, 包含电费回收编号、台区经理名称、台区经理工号、供电单位、欠费户数、欠费金额、电费回收率等字段, 其中电费回收编号是主键, 具体的表结构如表 4-6 所示:

表 4-6 电费回收率

编号	字段名	数据类型	非空	主键	注释
1	id	bigint(19)	是	是	电费回收编号
2	lob_manager_name	varchar(200)	是		台区经理名称
3	lob_manager_no	varchar(20)	是		台区经理工号
4	power_unit	varchar(200)	是		供电单位
5	overdue_no	int(11)	是		欠费户数
6	overdue_amt	varchar(20)	是		欠费金额
7	power_rate	varchar(20)	是		电费回收率
8	able_amt	varchar(20)	是		应收金额



## 4.3.2.3 安全生产模块

该模块主要是对工单、客户报修等信息进行管理,设计的数据表主要有工单、客户报修的信息表,下面是安全生产管理模块主要数据表的设计详情。

## (1) 工单表

工单表用于保存所有工单的记录情况,包含工单编号、执行单位、责任人、派单人、配合人、台区名称、派单地址、任务名称、客户名称、开单日期、派单日期、计划返回日期、工单类型、工单状态等字段,其中工单编号是主键,具体的表结构如表 4-7 所示:

表 4-7 工单表

编号	字段名	数据类型	非空	主键	注释
1	id	bigint(19)	是	是	工单编号
2	do_unit	varchar(20)	是		执行单位
3	charge_name	varchar(20)	是		责任人
4	sender_name	varchar(20)	是		派单人
5	cooperate_name	varchar(20)	是		配合人
6	lob_title	varchar(200)	是		台区名称
7	send_address	varchar(50)	是		派单地址
8	task_title	varchar(50)	是		任务名称
9	oprs_name	varchar(20)	是		客户名称
10	ord_date	datetime	是		开单日期
11	send_date	datetime	是		派单日期
12	plan_date	datetime	是		计划返回日期
13	actual_date	datetime	是		实际返回日期
14	ord_type	varchar(20)	是		工单类型
15	ord_status	varchar(20)	是		工单状态
16	ord_classify	varchar(20)	是		工单分类
17	power_unit	varchar(200)	是		供电单位
18	c_time	datetime	是		创建时间
19	u_time	datetime	是		更新时间
20	task_content	varchar(500)	是		任务内容

## (2) 客户报修表

客户报修表是统计辖区客户报修相关信息，包含报修编号、客户编号、电表局编号、客户名称、用电地址联系电话等字段，其中报修编号是主键，具体的表结构如表 4-8 所示：

表 4-8 客户报修表

编号	字段名	数据类型	非空	主键	注释
1	id	bigint(19)	是	是	报修编号
2	oprs_no	varchar(20)	是		客户编号
3	power_no	varchar(20)	是		电表局编号
4	oprs_name	varchar(20)	是		客户名称
5	power_address	varchar(200)	是		用电地址
6	phone	varchar(20)	是		联系电话
7	lob_no	varchar(20)	是		台区编号
8	lob_title	varchar(200)	是		台区名称
9	lob_manager_name	varchar(200)	是		台区经理名称
10	lob_manager_no	varchar(20)	是		台区经理工号
11	power_unit	varchar(200)	是		供电单位
12	repair_date	datetime	是		报修时间
13	repair_content	varchar(1000)	是		报修内容
14	c_time	datetime	是		创建时间

## 4.4 算法设计

算法根据一定的条件，对一些数据进行计算，从而得到需要的结果。算法是程序的灵魂，优秀的算法设计能显著提升程序的执行效率，产生事半功倍的效果<sup>[53]</sup>。算法设计过程包括选择合适的算法策略、详细规划算法执行的各个步骤，以及不断地对算法进行优化和完善。在整个程序开发过程中，算法设计不仅是核心要素，更是解决问题的关键方法。对于整个全能型供电所管理系统而言，有两个创新点，一个是智能自动派单，另一个是预测客户用电量。本小节将结合这两个创新点的实现，设计出指导开发的功能流程图，为后文中系统的开发实现提供业务框架基础。

#### 4.4.1 基于 TopN 和权重乘积之和的自动派单算法

基于 TopN 和权重乘积之和的自动派单算法分为五步，第一步，创建工单，将创建好的工单放进消息队列；第二步，监听消息队列，基于权重乘积之和算法计算每个接单人员的经纬度、天气、交通工具和交通拥挤度，使用 TopN 算法排行，将工单派发给前两名接单人员；第三步，判断派单次数是否超过 2 次，若超过 2 次则提示派单失败，若没有超过 2 次，锁定派单 90 秒，等待接单人员接单；第四步，若锁定派单 90 秒内有接单人员接单，则成功派单成功，使用 Redis 数据库锁住派单，派单任务结束，使用 Redis 数据库存储派单结果，再推送到前端显示；第五步，若锁定派单 90 秒内没有接单人员接单，上次计算出排名前两名接单人员作废，重新计算，派发给剩下接单人员中的前两名，继续循环执行第三步，直至派单任务结束。自动派单流程如图 4-11 所示：

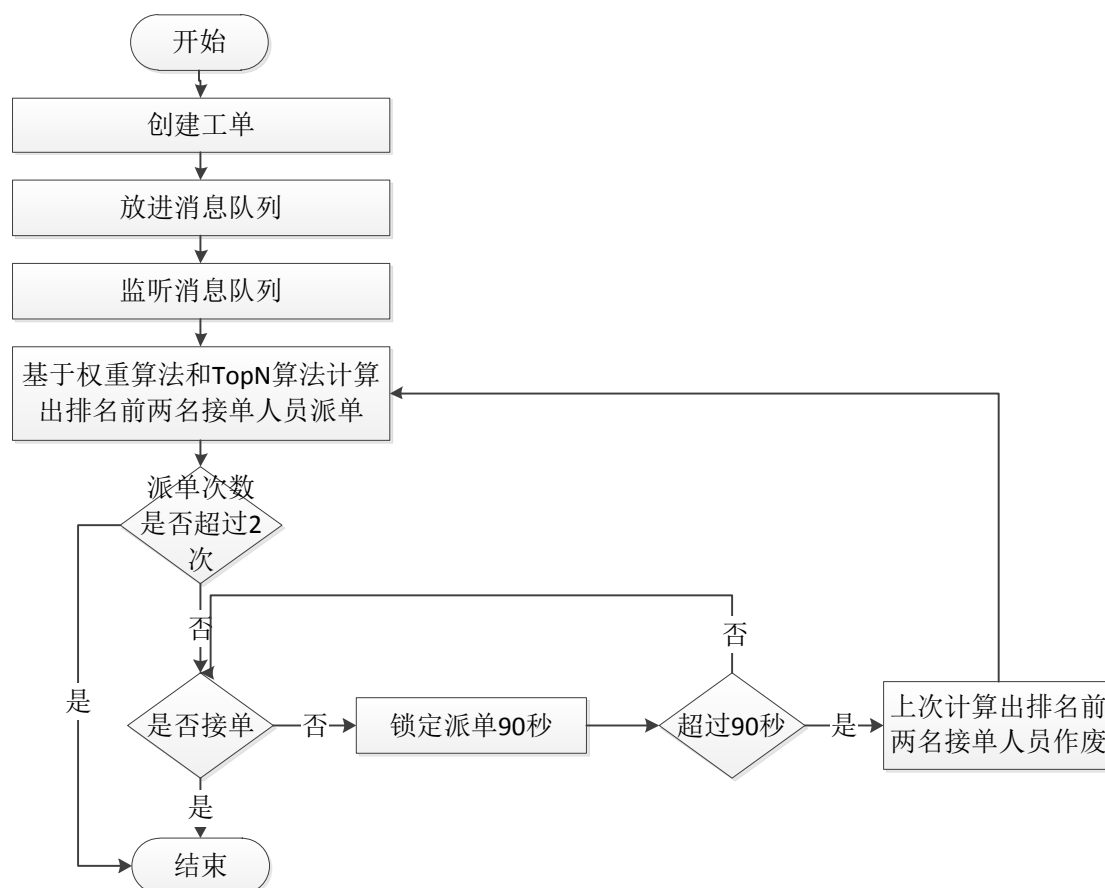


图 4-11 自动派单流程图

收集影响智能自动派单算法中的诸多参数，例如经纬度、天气、交通工具和交通拥挤度等，将每个参数划分为不同的因素。再根据权重乘积之和算法理论，计算每个接单人员不同因素的权重乘积和。

其具体计算函数如公式（4.1）所示：

$$S = w_1 \times x_1 + w_2 \times x_2 + \cdots + w_n \times x_n. \quad (4.1)$$

$w_i$ 对应于不同因素的权重值， $x_i$ 对应于不同因素的分值， $S$ 对应于每个接单人员不同因素的权重乘积和。

#### 4.4.2 基于最小二乘法的用电量预测算法

基于最小二乘法的用电量预测算法分为三步，第一步，基于最小二乘法模型对用户历史用电量数据进行拟合，形成预测数据；第二步，使用 Redis 数据库存储预测数据；第三步，前端使用 Echarts 库展示预测数据。预测客户用电量流程如图 4-12 所示：

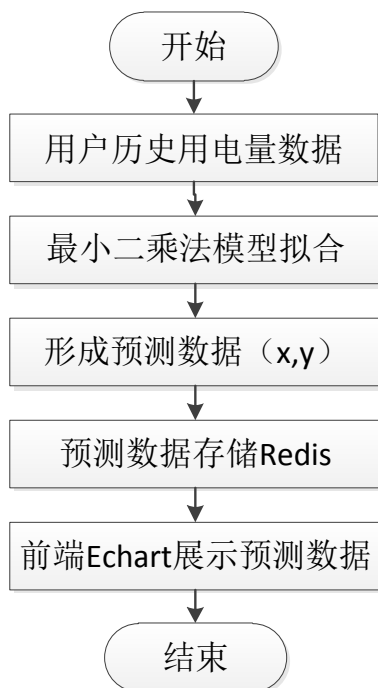


图 4-12 预测客户用电量流程图

在一般情况下，客户的用电习惯往往受到其生活方式、工作习惯、家庭结构等多种因素的影响，他们的用电习惯是有迹可循的。在预测客户用电量算法中，可以通过客户历史的用电量记录推算未来的用电量。客户历史的用电量由时间和

用电量两个因素组成，再根据最小二乘法进行线性拟合，最后得出用电天数与用电量的线性函数关系。

其具体计算函数如公式（4.2）和（4.3）所示：

$$a = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}. \quad (4.2)$$

$$b = \frac{1}{n} \sum_{i=1}^n y_i - \frac{a}{n} \sum_{i=1}^n x_i. \quad (4.3)$$

$$y = ax + b. \quad (4.4)$$

$x_i$ 对应于用电天数， $y_i$ 对应于用电量， $a$ 对应于截距， $b$ 对应于斜率。

## 4.5 本章小结

本章基于第三章的系统需求分析，在总体架构和技术架构上，将系统进行总的模块设计和详细的模块划分，设计了每个模块所涉及的数据表，完成了详细的数据库设计，对基于 TopN 和权重乘积之和的自动派单算法和最小二乘法的用电量预测算法进行了算法设计。

## 第五章 全能型供电所管理系统实现

本系统以供电所为研究对象,设计并实现了全能型供电所管理系统。系统使用 SpringBoot 和 Vue 框架,实现前后端分离。前端使用 Vue 框架和 Element-UI 相结合,后端使用 SpringBoot 框架编写独立的服务端代码。通过 ECharts 库绘制分布图,提供可视化数据展示。通过 MD5 算法实现用户数据的加密和解密,保护用户数据的安全性。实现基于 TopN 和权重乘积之和的自动派单算法和基于最小二乘法的用电量预测算法。采用 Shiro 安全机制,严格控制权限访问及系统安全,最终实现了满足实际需求的全能型供电所管理系统。

### 5.1 全能型供电所管理系统功能模块实现

#### 5.1.1 系统管理模块

在权限设置中,公司管理员具有使用系统管理模块的权限,因此仅有公司管理员账号能够看到系统管理模块菜单。主要实现了对各部门信息、角色权限、系统用户信息和系统菜单配置的管理功能。

##### (1) 部门信息管理

在部门信息管理模块中,由公司管理员维护并管理所有部门信息,修改所有部门信息,调整部门上下级关系,设置部门使用状态等功能。部门信息管理模块界面如图 5-1 所示:



图 5-1 部门管理模块界面

## （2）角色权限管理

角色管理模块将系统用户按角色分类，按角色分配权限，每个角色的系统权限不同。在角色管理模块，公司管理员依据系统及用户工作岗位实现系统用户角色的权限管理，修改所有角色信息，设置角色使用状态等功能。通过勾选角色授予权限，对该角色进行管控和约束。通过角色排序确定优先级，优先级越小，排名越靠前。角色管理模块界面如图 5-2 所示：

角色管理

+ 添加

删除

输入角色名

搜索

<input type="checkbox"/>	序号	角色名	状态	排序	操作
<input type="checkbox"/>	1	系统管理员	启用	1	<a href="#">授权</a> <a href="#">禁用</a>
<input type="checkbox"/>	2	海河供电所所长	启用	1	<a href="#">授权</a> <a href="#">禁用</a>
<input type="checkbox"/>	3	海河供电所营销员	启用	2	<a href="#">授权</a> <a href="#">禁用</a>
<input type="checkbox"/>	4	海河供电所核算员	启用	3	<a href="#">授权</a> <a href="#">禁用</a>
<input type="checkbox"/>	5	海河供电所安技员	启用	5	<a href="#">授权</a> <a href="#">禁用</a>
<input type="checkbox"/>	6	公司管理员	启用	4	<a href="#">授权</a> <a href="#">禁用</a>

图 5-2 角色管理模块界面

## （3）菜单管理

在菜单管理模块，公司管理员可以为不同角色的岗位指定系统菜单模块使用权限。通过菜单权限中选择部门角色，勾选对应权限，对该角色操作过程进行管理与控制，因此不同角色用户登录系统使用的菜单不同。菜单管理界面使用 ElementUI 框架中的树形组件显示菜单结构，并提供添加、删除、编辑和禁用的功能。菜单管理界面如图 5-3 所示：



\* 角色名

排序

菜单权限

- ☐ 综合管理
  - ☒ 所务公开
    - ☐ 基础信息维护
- ☐ 员工管理
  - ☐ 添加
  - ☐ 删除
  - ☐ 编辑
  - ☐ 禁用
- ☐ 系统管理
  - ☐ 用户管理
    - ☐ 添加

图 5-3 菜单管理界面

在后端 Controller 层中使用 Shiro 框架中的 @RequiresPermissions 注解, 引入 AOP 技术, 严格控制用户访问权限。根据登录用户编号查询菜单, 结合 Mybatis 框架完成菜单管理的增删改查等功能。Controller 层菜单管理关键代码如图 5-4 所示:

```
@RequiresPermissions("sys:menu:view")
@GetMapping("/{trees}")
public ApiResponse<?> getMenu() { return ApiResponse.ok(service.selectMenu(loginId: "0", isAdmin: 2)); }
@OperationLog(module = "系统菜单模块", type = "删除菜单", remark = "删除菜单")
@RequiresPermissions("sys:menu:delete")
@DeleteMapping("/{remove}/{ids}")
public ApiResponse<?> remove(@PathVariable String ids) {
    return service.deleteBatch(ids) > 0 ? ApiResponse.ok() : ApiResponse.error(); }
}
```

图 5-4 Controller 层菜单管理关键代码

### 5.1.2 工作看板模块

为更清晰了解各专业日常各项工作完成情况, 提高工作效率, 本系统为五类管理角色设置了不同的工作看板模块, 分别是公司管理员工作看板、所长工作看板、核算员工作看板、营销员工作看板和安技员工作看板。将重要的数据指标展示在工作看板上, 让数据实现可视化, 实现管理人员快速把握工作进度, 分析异常数据指标, 及时向相关部门、班组分配工作任务, 快速处理异常情况。公司管理员工作看板展示了分布图和整体指标完成情况。分布图包含各个供电所分布情况及三个指标完成情况。公司管理员工作看板界面如图 5-5 所示:

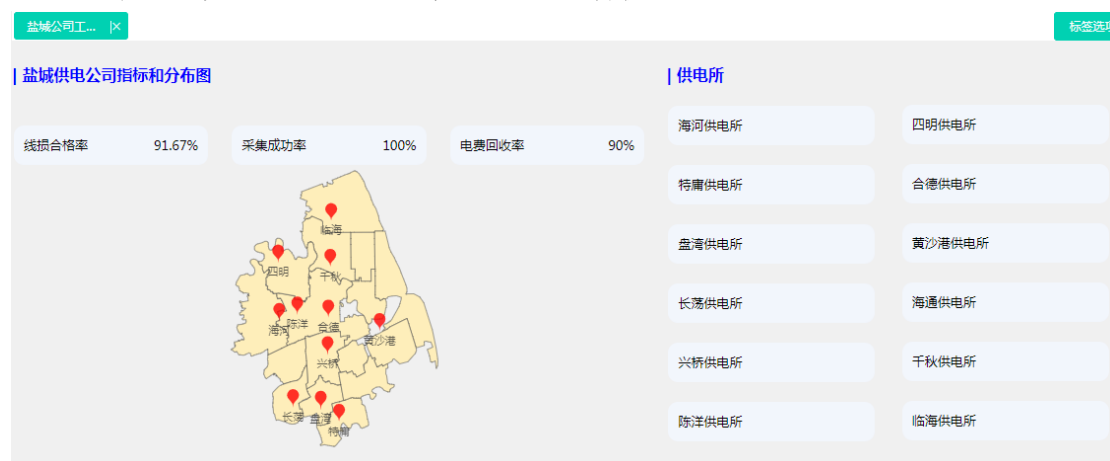


图 5-5 公司管理员工作看板界面



分布图在前端实现，使用 Echarts 库绘制，调用 Json 分布图的坐标数据。调取 Json 分布图坐标数据关键代码如图 5-6 所示：

```
init() {  
    echarts.registerMap( mapName: "盐城", JSON);  
    this.changeOptions( name: "盐城")  
    this.myChart = echarts.init(document.getElementById( elementId: "map"));  
    this.myChart.setOption(this.distributionOptions); },
```

图 5-6 调取 Json 分布图坐标数据关键代码

通过鼠标上的“滚轮”滚动，对分布图进行等比例缩放。图上用红色定位图标确定十二家供电所的地理坐标，点击图标查看所选供电所的三个关键指标（线损合格率、采集成功率和电费回收率）。当鼠标悬停在海河供电所上，详细显示了该供电所三个指标的完成情况，鼠标悬停在海河供电所上展示界面如图 5-7 所示：

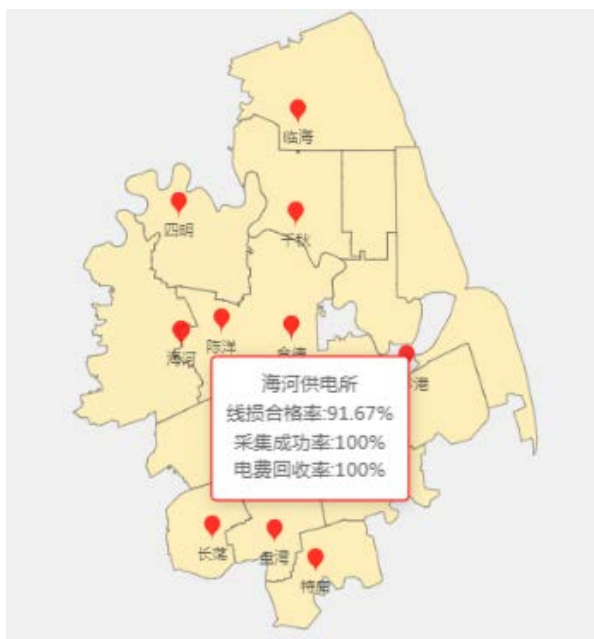


图 5-7 鼠标悬停在海河供电所上展示界面

将供电所管理人员按照不同岗位，划分为四类工作看板，易于各管理专业日常工作所需指标数据的统计与展示，使得管理人员更加高效地进行工作规划和任务分配。以所长工作看板为例，所长工作看板重点关注供电所内各管理专业的重点指标，如核算管理专业中的电费回收率、营销管理专业中线损合格率、安全生产中的用户低电压数等数据。所长工作看板界面如图 5-8 所示：



图 5-8 所长工作看板界面

以核算员工作看板为例，核算员工作看板显示供电所电费回收、逾期缴费统计等数据，能够第一时间通知台区经理处理并结清电费，提高总体电费回收率。电费回收率采用柱状图和折线图相结合，更直观地看出每位台区经理的电费回收情况。核算员工作看板界面如图 5-9 所示：



图 5-9 核算员工作看板

通过 Usercontoller 类的 getRoles 方法获取用户角色，Rolecontoller 类为不同的用户角色展示不同的工作看板，Usercontoller 类中关键代码如图 5-10 所示，Rolecontoller 类中关键代码如图 5-11 所示：

```

1  @RequiresPermissions("sys:user:view")
2  @PostMapping("/list")
3  public ApiResponse<?> getList(@RequestBody Search dto) { return ApiResponse.ok(userService.userList(dto)); }

4  @RequiresPermissions("sys:user:view")
5  @GetMapping("/perms/{id}")
6  public ApiResponse<?> getPerms(@PathVariable String id) {
7      return ApiResponse.ok(userService.selectMenuPerm(id, userService.isAdmin(id)));
8  }

9  @RequiresPermissions("sys:user:view")
10 @GetMapping("/roles/{id}")
11 public ApiResponse<?> getRoles(@PathVariable String id) {
12     List<String> list = userService.selectRoles(id);
13     return ApiResponse.ok(list);
14 }

```

图 5-10 Rolecontoller 类中关键代码

```

1  @RequiresPermissions("sys:role:view")
2  @PostMapping("/list")
3  public ApiResponse<?> getList(@RequestBody JSONObject body) {
4      return ApiResponse.ok(service.selectRoleList(body.getString( key: "keyword"),
5          body.getInteger( key: "index"),
6          body.getInteger( key: "size")));
7  }
8  @RequiresPermissions("sys:role:view")
9  @GetMapping("/detail/{id}")
10 public ApiResponse<?> getDetail(@PathVariable String id) { return ApiResponse.ok(service.selectRoleData(id)); }

```

图 5-11 Usercontoller 类中关键代码

### 5.1.3 指标看板模块

五类角色指标看板模块内容一致，展示供电所在生产运维、营销服务和综合管理上的重要指标，以及指标在全县排名和供电所指标最优值。指标通过后端读取数据库，再通过前端展示。供电可靠率指是供电质量的重要指标，《国家电网公司供电服务“十项承诺”》中规定：城市电网平均供电可靠率达到 99.9%，乡镇电网平均供电可靠率达到 99.8%。万户故障报修率指（万户中报修户数量/万户数量）\*100%。电费回收率=实收电费/应收电费\*100%，实收电费=应收电费-欠费金额。线损合格率=（线损合格台区数/总台数）\*100%。采集成功率=（召测到数据电表数/实际电表数）\*100%。指标看板模块界面如图 5-12 所示：

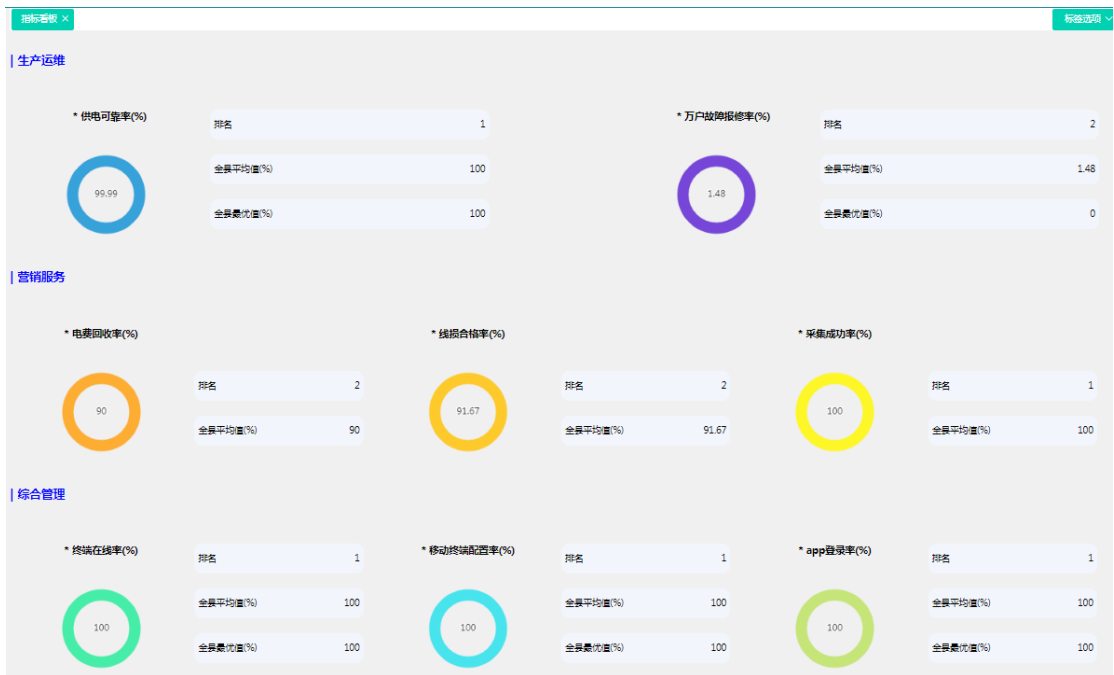


图 5-12 指标看板模块界面

5.1.4 营销服务模块

营销服务模块有利于营销员角色查看台区线损，并对异常线损的损失电量进行分析。通过输入“单位”、“台区经理”、“线损类别”等信息查询相关台区线损情况。线损率=（台区供电量-台区售电量）/台区供电量。线损量即线路损失电量，线损量=台区供电量-台区售电量。台区线损合格率界面如图 5-13 所示：

台区线损合格...											标签选择
输入用电单位		输入台区经理名称		选择线损合格分类	选择日期						搜索
<input type="checkbox"/>	序号	台区编号	台区名称	台区经理名称	台区经理工号	供电量	售电量	线损量	线损率	线损分类	供电单位
<input type="checkbox"/>	1	309****4669	施河陡****5#台区	袁一	10899475	334.8	329.32	5.48	1.636798	正常	施河供电所
<input type="checkbox"/>	2	099****3634	阜余大****8#台区	陈二	10841594	249.39	245.01	4.38	1.756285	正常	施河供电所
<input type="checkbox"/>	3	099****3567	阜余大****2#台区	张三	10979905	342.4	339.28	3.12	0.911215	正常	施河供电所
<input type="checkbox"/>	4	099****5523	阜余新****1#台区	李四	10898097	191.96	188.61	3.35	1.745155	正常	施河供电所
<input type="checkbox"/>	5	099****4987	施河陡****1#台区	苏五	10898287	171.54	167.74	3.8	2.215227	正常	施河供电所
<input type="checkbox"/>	6	099****5101	施河陡****8#台区	韩六	10899051	191.2	185.88	5.32	2.782427	正常	施河供电所
<input type="checkbox"/>	7	309****3561	阜余大****9#台区	王七	10899347	223.21	217	6.21	2.782133	正常	施河供电所

图 5-13 台区线损合格率界面

将台区线损类别形成台区线损汇总，能够直观统计和查询。线损类别分为负损、正常和高损。正常为合格线损，负损和高损为不合格线损。合格线损区间为

$[-0.5, 3]$ ，小于-0.5 为负损，大于 3 是高损。台区线损汇总界面如图 5-14 所示：

<input type="checkbox"/>	序号	台区经理名称	台区经理工号	运行台数	负损	正常	高损	合格区间	不合格区间	线损合格率	供电单位
<input type="checkbox"/>	1	夏一	10899475	2	0	2	0	2	0	100	鹿河供电所
<input type="checkbox"/>	2	陈二	10841594	3	0	3	0	3	0	100	鹿河供电所
<input type="checkbox"/>	3	张三	10979905	1	0	1	0	1	0	100	鹿河供电所
<input type="checkbox"/>	4	李四	10898097	2	0	2	0	2	0	100	鹿河供电所
<input type="checkbox"/>	5	苏五	10898287	3	1	2	0	2	1	66.67	鹿河供电所
<input type="checkbox"/>	6	韩六	10899051	1	0	1	0	1	0	100	鹿河供电所
<input type="checkbox"/>	7	王七	10899347	2	0	1	1	1	1	50	鹿河供电所

图 5-14 台区线损汇总界面

台区线损汇总按线损类别将台区线损合格率以数值的形式汇总显示，通过点击数值链接跳转筛选后的台区线损情况。通过 Mybatis 框架中的 RowBounds 类实现分页，一个页面放 7 条数据。台区线损汇总关键代码如图 5-15 所示：

```
public ApiPage<LobPowerSum> list(PageReq pageReq) {  
    Example example = new Example(LobPowerSum.class);  
    RowBounds rowBounds = new RowBounds( offset: (pageReq.getIndex() - 1) * pageReq.getSize(), pageReq.getSize());  
    return new ApiPage<>(pageReq.getSize(),  
        pageReq.getIndex(),  
        lobPowerSumMapper.selectCountByExample(example),  
        lobPowerSumMapper.selectByExampleAndRowBounds(example, rowBounds));  
}
```

图 5-15 台区线损汇总关键代码

### 5.1.5 核算管理模块

核算员通过核算管理模块中的电费明细和欠费明细告知台区经理所辖用户电费缴费情况，台区经理通知客户缴费，及时完成电费回收指标。欠费明细是在电费明细的基础上添加筛选的功能，可根据“供电单位”、“催费日期”、“催费责任人”和“欠费状态”（欠费或者正常）进行筛选。欠费明细界面如图 5-16 所示：

电费明细 1x

标签选择

输入供电单位

输入缴费责任人

日期选择日期

欠费

搜索

生成工单

导出Excel

<input type="checkbox"/>	序号	用户编号	人员名称	用电地址	联系电话	台区编号	台区名称	台区经理姓名	台区经理工号	供电单位	应收电量	应付金额	实收金额	欠费金额	缴费渠道
<input type="checkbox"/>	1	740****9053	新***变	皇余***3组	138****5066	309****4669	海河路****5#台区	夏一	10899475	海河供电所	153	81	0	81	
<input type="checkbox"/>	2	710****6161	和***变	海河****河堤	137****8833	099****3634	皇余大****8#台区	陈二	10841594	海河供电所	280	147	8	139	
<input type="checkbox"/>	3	740****6735	海****区	海河****六组	136****9298	099****3567	皇余大****2#台区	张三	10979905	海河供电所	224	118	0	118	
<input type="checkbox"/>	4	710****4829	陆***变	新河****河堤	138****3299	099****5523	皇余新****1#台区	李四	10898097	海河供电所	551	265	17	248	
<input type="checkbox"/>	5	740****3232	皇***变	海河****胜村	139****0173	099****4987	海河海****1#台区	郑五	10898287	海河供电所	376	198	0	198	
<input type="checkbox"/>	6	740****0198	大****区	海河****日村	139****2008	099****5101	海河海****8#台区	韩六	10899051	海河供电所	155	79	0	79	
<input type="checkbox"/>	7	740****0088	新****区	海河****委会	138****0299	309****3561	皇余大****9#台区	王七	10899347	海河供电所	485	256	0	256	

<

1

2

3

4

5

6

...

72

>

图 5-16 欠费明细界面

使用 exportExcel 方法，定义 Excel 属性信息、文件头信息以及 Sheet 名称，使用 IO 流将数据库中的电费数据导出成 Excel 表格，同时捕获导出 Excel 表格失败的异常。将电费数据导出为 Excel 表格的形式并打印成纸质版，便捷地将电费数据呈现给台区经理查看。导出电费明细关键代码如图 5-17 所示：

```
public static void exportExcel(Collection<?> listData, Class<?> pojoClass, String headTitle, String sheetName,
    HttpServletResponse response) {
    ExportParams params = new ExportParams(headTitle, sheetName);
    params.setHeight((short) 8);
    params.setStyle(ExcelStyle.class);
    try {
        Workbook workbook = cn.afterturn.easypoi.excel.ExcelExportUtil.exportExcel(params, pojoClass, listData);
        String fileName = headTitle + new SimpleDateFormat( pattern: "yyyyMMddHHmmss").format(new Date());
        fileName = URLEncoder.encode(fileName, StandardCharsets.UTF_8);
        response.setContentType("application/vnd.ms-excel; charset=utf-8");
        response.setHeader( s: "Content-Disposition", s1: "attachment; filename=" + fileName + ".xls");
        ServletOutputStream out = response.getOutputStream();
        workbook.write(out);
        out.flush();
        out.close(); } catch (Exception e) {
    logger.error("导出excel报错:", e); } }
```

图 5-17 导出电费明细关键代码

通过勾选欠费信息，一键生成欠费工单，采用 Redis 数据库做组件发送给指定台区经理账号，提醒台区经理及时完成工单任务。使用雪花算法生成 19 位工单编号，保证了工单编号的唯一性。生成工单关键代码如图 5-18 所示：

```
if (null == workOrder.getId()) {  
    workOrder.setId(String.valueOf(snowflakeUtil.nextId()));  
    workOrder.setOrdNo(String.valueOf(snowflakeUtil.nextId()));  
    workOrder.setCTime(new Date());  
    workOrder.setUTime(new Date());  
    workOrder.setOrdDate(DateFormatUtil.formatTime(new Date()));  
    int insert = workOrderMapper.insert(workOrder);  
    stringRedisTemplate.convertAndSend(channel: "work." + workOrder.getId(), JSONObject.toJSONString(workOrder));  
    return insert;  
}
```

图 5-18 生成工单关键代码

5.1.6 工单管理模块

在工单管理模块,通过根据“单位”、“工单编号”、“工单责任人”、“派单时间”、“派单人”、“工单分类”、“工单状态”等筛选已经生成的工单。选中工单记录,实现批量删除和单个删除的功能,若不选中工单记录或者删除时会弹出提醒。除新增工单外,可在电费明细和故障抢修模块中生成工单,由登录账号进行派单,派单给台区经理完成。工单管理模块界面如图 5-19 所示:

工单管理 X

+ 添加

删除

执行单位

工单编号

工单责任人

派单人

搜索

自 开单日期

工单分类

工单状态

<input type="checkbox"/>	序号	执行单位	工单编号	责任人	派单人	配合人	台区名称	派单地址	任务名称	派单日期	计划返回日期	实际返回日期	工单状态	工单分类	供电单位	操作
<input type="checkbox"/>	1	海河供电所	15644418 18081669 121	李四	封阳公司管 理员		海河路**** 5#台区	奥余****3 组	电费管理	2022-08-2 8 00:00:00	2022-09-2 9 00:00:00	2022-09-2 9 00:00:00	已完成	电费管理	海河供电所	<a href="#">处理</a> <a href="#">终止</a>
<input type="checkbox"/>	2	海河供电所	15619138 15367589 889	李四	封阳公司管 理员	1	海河路**** 5#台区	随变	采集维护	2022-08-2 6 00:00:00	2022-08-2 7 00:00:00	2022-08-2 8 00:00:00	已超时	采集维护	海河供电所	<a href="#">处理</a> <a href="#">终止</a>
<input type="checkbox"/>	3	海河供电所	15605139 42445768 705	李四	封阳公司管 理员		海河路**** 5#台区	随变	采集维护	2022-08-0 2 00:02:02	2022-08-2 5 00:00:00	2022-08-2 8 00:00:00	已超时	采集维护	海河供电所	<a href="#">处理</a> <a href="#">终止</a>

<

1

>

图 5-19 工单管理模块界面

工单涉及的更改、新增、删除操作统一通过 AOP 模式拦截请求,并将每次更改的操作日志记录到数据库中。添加 Maven 依赖,在 Pom.xml 中引入响应的依赖模块,再新建自定义注解类。通过将切面类注入容器,在类的前面分别加上注解@Aspect 和@Component。AOP 模式拦截请求关键代码如图 5-20 所示:



```
@Aspect
@Component
public class OperationLogAspect {
    private static final Logger logger = LoggerFactory.getLogger(OperationLogAspect.class);
    @Resource
    private OperationLogService operationLogService;
    @Resource
    private SnowflakeUtil snowflakeUtil;
    @Pointcut("@annotation(com.vane.system.annotation.OperationLog)")
    public void operationLogPointCut() {
    }
}
```

图 5-20 AOP 模式拦截请求关键代码

5.1.7 安全生产模块

在安全生产模块的故障抢修子模块中，详尽地记录了每个客户的基本信息以及他们对应的台区经理。当安技员接收到客户的故障报修时，他们会在安全生产模块中的故障抢修子模块中详细记录报修内容和报修时间。这一记录随后可以生成一个相应的工单，并立即通知相应的台区经理前往客户家中处理故障。从而缩短故障修复时间，提升客户满意度。工单的完成时间即为客户故障抢修结束的时间，通过“客户编号”、“抢修开始时间”和“结束时间”进行详细筛选，使得安技员能够迅速定位并获取所需的客户信息，以便进行后续的故障分析和优化工作。故障抢修模块界面如图 5-21 所示：

故障报修	新增											标签选择	
+ 添加		输入供电单位				输入用户编号		输入报修开始时间		输入报修结束时间		搜索	生成工单
<input type="checkbox"/>	序号	用户编号	电表编号	人员名称	用电地址	联系电话	台区编号	台区名称	台区经理名称	台区经理工号	供电单位	报修时间	报修内容
<input type="checkbox"/>	1	740****9053	053****4230	靳****亮	皇余****3组	138****5066	309****4669	海河庭****5#台区	袁一	10899475	海河供电所	2022-08-23 00:00:00	家中没电
<input type="checkbox"/>	2	740****9053	053****4230	靳****亮	皇余****3组	138****5066	309****4669	海河庭****5#台区	袁一	10899475	海河供电所	2022-07-25 00:02:02	家中没电
<input type="checkbox"/>	3	710****1728	100****7824	杨****青	海河****委会	138****4603	099****5101	海河庭****8#台区	韩六	10899051	海河供电所	0000-00-00 00:00:00	
<input type="checkbox"/>	4	710****6538	100****7823	廖****中	皇余****康村	138****1444	099****4987	海河庭****1#台区	苏五	10898287	海河供电所	0000-00-00 00:00:00	
<input type="checkbox"/>	5	746****9969	100****7822	陈****付	海河****新村	137****6498	099****5523	皇余新****1#台区	李四	10898097	海河供电所	0000-00-00 00:00:00	
<input type="checkbox"/>	6	746****4917	100****7791	永****业	海河****委会	135****4839	099****3567	皇余大****2#台区	张三	10979905	海河供电所	0000-00-00 00:00:00	
<input type="checkbox"/>	7	746****4629	100****7787	路****助	皇余****台区	139****2127	099****3634	皇余大****8#台区	陈二	10841594	海河供电所	0000-00-00 00:00:00	
<	1	2	3	4	5	6	...	72	>				

图 5-21 故障抢修模块界面



5.1.8 综合管理模块

所务公开模块作为一个供电所信息展示页面，集中汇总了基础管理信息和员工管理两个子模块的核心内容。通过这一模块，其他角色能够便捷地获取供电所的详细信息和员工信息，从而增进对供电所基本管理和员工管理状况的了解。所务公开模块界面如图 5-22 所示：



图 5-22 所务公开模块界面

为保证信息的准确性，仅设置了所长和公司管理员角色能够修改基础管理信息模块和员工管理模块的信息。支持输入“用户名”、“手机号”等信息查询相关用户，勾选相应用户，设置其使用状态。员工管理模块界面如图 5-23 所示：



图 5-23 员工管理模块界面

5.2 辅助全能型供电所管理系统功能模块实现

辅助全能型供电所管理系统功能模块包含登录注册功能模块和菜单界面功能模块。

### 5.2.1 登录注册模块

登录注册模块是一个系统最基本的功能,在前端 Vue 项目中引入 Element-UI 组件实现登录注册界面,在用户初次使用系统前,需要先注册系统,在注册界面中,根据文字提示,输入用户的注册信息,选择“供电单位”,若账户名、密码和确认密码不规范则会出现 Element-UI 组件 Message 报错提醒。注册成功后,弹出 Element-UI 组件 Message “注册成功”提示信息,通过注册路由跳转到登录界面。注册信息注册界面如图 5-24 所示:



图 5-24 注册界面

用户第一次注册时,前端对用户的注册信息先使用 MD5 算法进行加密,再发送传给存储数据库,保障了用户信息的安全性。MD5 算法加密关键代码如图 5-25 所示:

```
public int saveUser(JSONObject json) {  
    Example e = new Example(User.class);  
    e.and().andCondition("login_id='" + json.getString( key: "loginId") + "'");  
    List<User> users = userMapper.selectByExample(e);  
    if (!EncryptUtil.md5(json.getString( key: "oldPwd")).equals(users.get(0).getPassword())) {  
        return 2;  
    }  
    return userMapper.updatePwd(json.getString( key: "loginId"),  
        EncryptUtil.md5(json.getString( key: "newPwd")));  
}
```

图 5-25 MD5 算法加密关键代码

在登录界面中,根据文字提示,输入已注册的登录名和密码,输入验证码点击登录。当输入登录信息时,需要验证用户名、密码和验证码,若用户输入的密码与数据库中密码通过 MD5 解密后相一致,才能成功登录。登录成功后,路由跳转到系统主页面,并出现“登录成功!”的 Element-UI 组件 Message 消息提示,登录界面如图 5-26 所示:

图 5-26 登录界面

验证码由数字和英文字母组成,以图片形式显示,点击验证码自动刷新,后端生成验证码,存储到 Redis 数据库,返回验证码给前端页面展示。用户输入验证码,点击登录按钮提交验证,后端将用户输入的验证码进行校验,对比用户输入的验证码是否和后端生成的一致,一致则验证成功,否则验证失败。

点击“忘记密码”路由跳转到“重置密码”的界面,进行密码重置功能。“记住密码”功能是将用户名和密码采用 Base64 加密,并保存在 Redis 数据库中。

记住密码功能关键代码如图 5-27 所示:

```
if (this.checked) {
  this.setCookie( cName: "account", this.ruleForm.loginId)
  let passWord = Base64.encode(this.ruleForm.password)
  this.setCookie( cName: "password", passWord)
} else {
  this.setCookie( cName: "account", value: "")
  this.setCookie( cName: "password", value: "")
}
this.$router.push('/');
return;
```

图 5-27 记住密码功能关键代码

登录成功的用户，通过点击修改个人信息菜单，可以查看和修改用户的个人信息数据，修改后的数据直接存储到数据库中。用户信息存储到数据库前，需要通过 Axios 进行请求拦截器和响应拦截器设置。在 Axios.js 文件中对 Axios 进行配置，请求拦截器对 Http 请求进行统一转换和处理请求头信息，目的是统一配置请求信息及错误处理。Axios 请求拦截器的关键代码如图 5-28 所示：

```
axios.interceptors.request.use(  
  onFulfilled: config => {  
    let obj = signature.genSignature(config.url, config.data, config.method)  
    for (let key in obj) {  
      if (key === 'secretKey') continue  
      config.headers[key] = obj[key]  
    }  
    return config;  
  },  
  onRejected: err => {  
    return Promise.reject(err);  
  }  
);
```

图 5-28 Axios 请求拦截器的关键代码

响应拦截器是在数据库收到数据时，进行数据过滤、对状态码判断，进行对应的操作。如在数据库返回登录状态失效时，需要用户重新登录，系统将会弹出“登录状态失效”的提示信息，并跳转到相应登录页面。Axios 响应拦截器的关键代码如图 5-29 所示：

```
axios.interceptors.response.use(  
  onFulfilled: res => {  
    if (res.data.code === 10006) {  
      localStorage.removeItem(key: 'user');  
      localStorage.removeItem(key: 'sessionId');  
      router.push({  
        path: "/login"  
      });  
    }  
    return res;  
  }  
);
```

图 5-29 Axios 响应拦截器的关键代码

### 5.2.2 菜单界面模块

每个用户的角色不同，本系统根据用户角色分配了不同的权限，因此展示的菜单界面也不同。用户角色分为五类，分别是公司管理员、所长、核算员、营销员和安技员。公司管理员菜单界面如图 5-30 所示，其他角色菜单展示内容由公司管理员控制。所长菜单界面如图 5-31 所示：



图 5-30 公司管理员菜单界面



图 5-31 所长菜单界面

由于每个菜单展示内容过多，设置了主页面全屏展示和菜单折叠功能，菜单折叠前和折叠后的界面如图 5-32 所示和 5-33 所示：



图 5-32 菜单折叠前界面



图 5-33 菜单折叠后界面

菜单默认不折叠，点击折叠图标，实现菜单折叠，使得系统主使用界面更加简洁大方。折叠通过 Event Bus 进行组件间通信，折叠侧边栏，菜单折叠功能关键代码如图 5-34 所示：

```
data() {  
  return {  
    collapse: false,  
    items: menu.state.navTree,  
  }  
},  
computed: {  
  onRoutes() {  
    return this.$route.path.replace('/', '');  
  },  
},  
methods: {  
},  
created() {  
  // 通过 Event Bus 进行组件间通信，来折叠侧边栏  
  bus.$on('collapse', msg => {  
    this.collapse = msg;  
  })  
},  
mounted() {
```

图 5-34 菜单折叠功能关键代码

## 5.3 算法实现

### 5.3.1 基于 TopN 和权重乘积之和的自动派单算法

#### 5.3.1.1 数据准备

智能自动派单的实现需要考虑诸多参数，主要有经纬度、天气、交通工具和交通拥挤度等。这里根据权重乘积之和算法理论，将这些参数再详细划分不同因素，将这些因素赋分值（ $x_i$ ）和权重值（ $w_i$ ），计算出的乘积之和（ $S$ ）越大，排名越靠后。详细参数如表 5-1 所示：

表 5-1 详细参数表

参数	因素	分值 (xi)	权重值 (wi)
经纬度	距离	按距离动态计算	0.6
	晴天	100	0.1
天气	雨天	200	0.1
	雪天	300	0.1
	汽车	100	0.1
交通工具	骑行	200	0.1
	步行	300	0.1
交通拥挤度	拥挤	200	0.2
	不拥挤	100	0.2

### 5.3.1.2 筛选排行

在工单自动派发给当前合适的接单人员过程中,应先计算他们的排名,再选择前两名接单人员进行工单派发。基于 TopN 算法对计算出的乘积之和 (S) 排序,从而得到接单人员的排名。

### 5.3.1.3 代码实现

智能自动派单的确定需要考虑诸多参数,将经纬度、天气、交通工具和交通拥挤度等参数赋分值和权重值。当工单创建后,将经纬度、天气、交通工具和交通拥挤度等参数赋分值和权重值后放入 Redis 消息队列,在取出时进行根据权重乘积之和算法计算。权重乘积之和算法是计算权重 (values\_w) 和值 (values\_v) 两个列表的加权和。每个参数的权重与其对应的值相乘,再将所有乘积相加得到最终的加权和 (weighted\_sum)。权重乘积之和算法伪代码如下:

---

#### Algorithm 1 Weighted Sum of Products Algorithm

---

**Input:** A list of values\_w, a list of values\_v. //权重与值一一对应,长度均为 n

**Output:** An integer weighted\_sum.

```

1  weighted_sum = 0;
2  for (i=1; i≤n; i++) do
3      weighted_sum ← weighted_sum + w[i] * v[i];
4  end for
5  return weighted_sum;
```

---

基于 TopN 算法排序，在所有的台区经理中筛选出前两名接单人员进行派单和消息推送。TopN 算法先对列表 (L) 进行降序排序，最大元素将排在列表 (L) 的前面。初始化一个空列表 (TopItems)，用于存储最终选出最高的 (N) 个项。通过索引 (i) 遍历排序后列表 (L) 的前 (N) 个元素，并将它们逐个添加到列表 (TopItems) 中，最后返回列表 (TopItems) 中的最高 (N) 个项。TopN 算法伪代码如下：

---

**Algorithm 2** TopN Algorithm
 

---

**Input:** A list L of items, an integer N. //要找到的最高项数

**Output:** A list TopItems of items. //从 L 中选出的 N 个最高项

```

1  function TopN(L, N)
2      Sort the list L in descending order;
3      TopItems = [ ]; //初始化空列表
4      if (N > length(L)) then
5          N = length(L); //确保 N 不超过列表 L 的长度
6      end if
7      for (i=0; i<N; i++) do
8          TopItems.append(L[i]);
9      end for
10     return TopItems;
11 end function
  
```

---

在系统界面，能够实时查看智能派发工单后的接单情况，当派发的接单人员登录系统时，会收到是否接单提醒，实时查看接单情况界面如 5-35 所示：

编辑

执行单位	海河供电所	责任人	请选择
派工人	射阳公司管理员	配合人	
台区名称	海河陡****5#台区	* 派单地址	阜余****3组
* 任务名称	电费工单	工单分类	电费管理
任务内容		派单日期	③
计划返回日期	③	实际返回日期	③

处理人	状态	时间
接单人员4	未接收	2024-01-29 19:47:18
接单人员5	未接收	2024-01-29 19:47:18

提交处理
处理完成

图 5-35 实时查看接单情况界面



5.3.2 基于最小二乘法的用电量预测算法

5.3.2.1 数据准备

本次随机选取海河供电所 2024 年 1 月 1 日至 1 月 10 日 5000 户居民客户用电量记录。由于用电量与客户用电天数之间存在关联关系，算出同一时间段客户日用电量的平均值，最后得出用电天数与用电量的关系。5000 户居民客户 10 日用电量详细参数如表 5-2 所示：

表 5-2 5000 户居民客户用电量记录表

部门	用电量 (xi)	时间 (yi)
海河供电所	66156	2024-01-01
海河供电所	76537	2024-01-02
海河供电所	84235	2024-01-03
海河供电所	86874	2024-01-04
海河供电所	98965	2024-01-05
海河供电所	92488	2024-01-06
海河供电所	104256	2024-01-07
海河供电所	81521	2024-01-08
海河供电所	92516	2024-01-09
海河供电所	109564	2024-01-10

5.3.2.2 线性拟合

采用最小二乘法构建线性回归模型的拟合原理，依据表 5-2 的数据给出xi和yi两个坐标对应的两个数组值。xi是用电天数，yi是用电量，对两组数据基于最小二乘法进行线性拟合，得出近似函数 $y = ax + b$ ，从而预测出 1 月 11 日至 1 月 15 日的用电量。最后得出的xy散点图及线性拟合结果如图 5-36 所示：

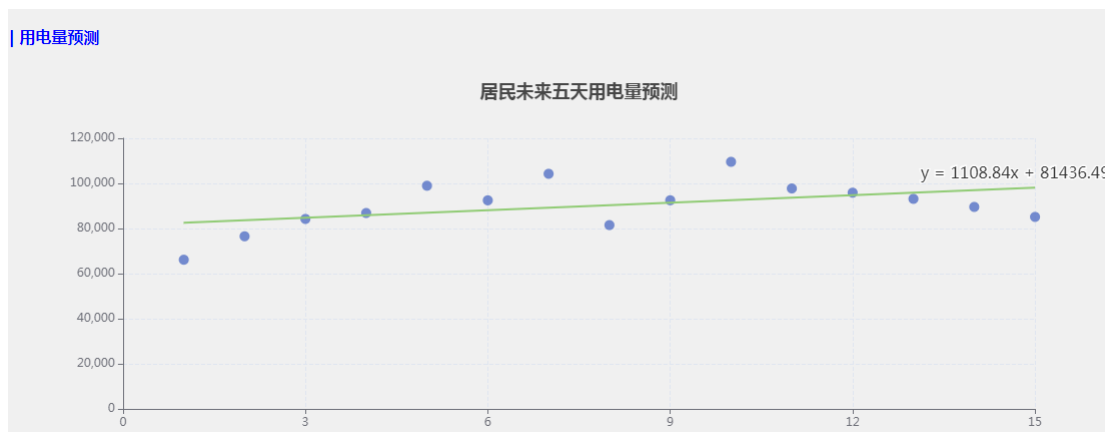


图 5-36 xy散点图及线性拟合图

因此，用 $y = 1108.84x + 81436.49$ 作为这 5000 户居民用户的用电趋势。

### 5.3.2.3 实验对比

本节根据 5000 户居民客户 2024 年 1 月 1 日至 1 月 10 日用电量，将 1 月 11 日至 1 月 15 日的实际用电量与预测 1 月 11 日至 1 月 15 日用电量做实验对比。采用真实用电量形成平滑曲线图，基于最小二乘法构建线性回归模型预测未来 5 天用电量形成线性拟合图，通过平滑曲线图和线性拟合图进行对比，实验对比图如图 5-37 所示：

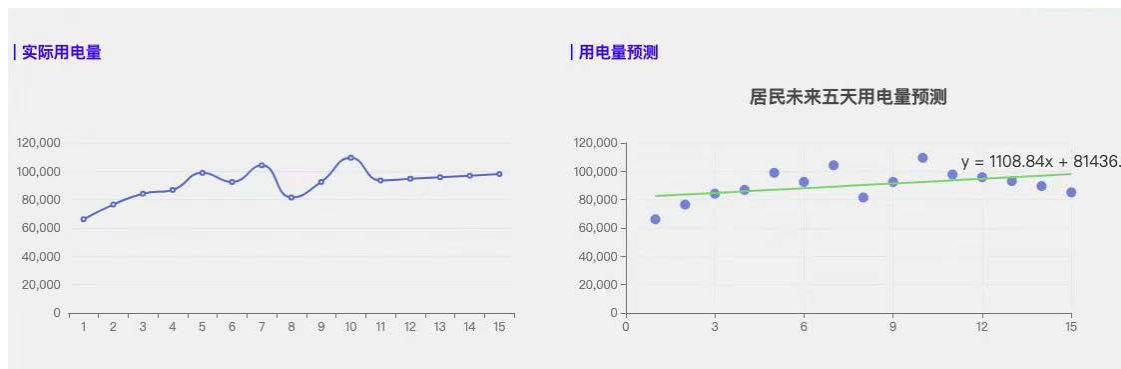


图 5-37 实验对比图

通过对比两个实验所生成的对比图，数据趋势基本一致，这一结果有力地证明了基于最小二乘法构建的线性回归模型在预测选取的 5000 户居民客户电量方面的准确性和可靠性，从而提前预知客户用电趋势，帮助客户减少用电成本。

### 5.3.2.4 代码实现

通过对客户用电量进行预测，采用最小二乘法算法构建线性回归模型，通过分散的样本点得到自变量（天数 $x$ ）和因变量（用电量 $y$ ）之间的关系，形成一条 $y = ax + b$ 的线性函数。首先计算天数（ $x$ ）和用电量（ $y$ ）的平均值（ $\text{mean}_x$ ）和（ $\text{mean}_y$ ），再通过遍历所有数据点，计算斜率（ $a$ ）的值，最后根据斜率（ $a$ ）和（ $x$ ）值的平均值计算截距（ $b$ ）。最小二乘法算法伪代码如下：

---

**Algorithm 3** Least Squares Method Algorithm

---

**Input:** A list of values<sub>x</sub>, a list of values<sub>y</sub>. //长度均为 n

**Output:** A number a, a number b.

```
1  mean_x = Mean(values_x); //计算 x 的均值
2  mean_y = Mean(values_y);
3  numerator = 0; //初始化分子
4  denominator = 0;
5  for (i=1; i≤n; i++) do
6      numerator = numerator + (values_x[i] - mean_x) * (values_y[i] - mean_y);
7      denominator = denominator + (values_x[i] - mean_x)^2;
8  end for
9  a = numerator / denominator; //计算斜率 a
10 b = mean_y - a * mean_x; //用斜率 a 计算截距 b
11 return a, b;
```

---

## 5.4 本章小结

本章内容在第三章系统需求分析和第四章系统详细设计的基础上，详细描述了系统每个功能模块的具体实现，对主要功能界面进行了展示，并介绍了关键功能模块的部分代码和算法。

## 第六章 系统测试

系统测试是项目研发的检验环节，主要对功能、性能等方面进行测试用于保证开发质量<sup>[54]</sup>。测试的核心目的在于确保系统可以稳定可靠地运行，进而最大限度地减少系统出现错误或故障的可能性，从而提升用户体验和系统的整体性能<sup>[55]</sup>。系统测试能够测试开发的系统是否存在问题，并解决问题。这些问题包含系统的功能性和非功能性问题，整个系统测试贯穿系统开发全过程，只有经过无数次的系统测试和问题修改，开发后的系统才能正式投入使用。通过不断地系统测试能减少系统故障发生次数，提高系统的实用性，增强用户的体验度。本章主要对全能型供电所管理系统环境、功能性和非功能性模块展开测试。

### 6.1 系统测试环境

系统测试环境：

- (1) 操作系统：Windows 8.1
- (2) 服务器：Tomcat 8.0
- (3) 数据库：MySQL 8.0.15、Redis 3.2
- (3) 数据库管理工具：Navicat 15
- (4) 开发环境：IntelliJ IDEA 2023
- (5) 浏览器：Google Chrome、360 极速浏览器、Firefox

### 6.2 功能性测试

功能测试主要用于验证一个项目的功能需求是否达到用户提出的预期标准，测试前需要设计测试用例<sup>[56]</sup>。测试用例的设计必须严谨且完整，每个用例都应逐一进行测试，以确保系统功能的全面覆盖。规范的测试用例需要识别系统中潜在的缺陷，解决这些缺陷问题后，能有效降低系统的出错率，提高系统的整体性能<sup>[57]</sup>。设计测试用例分为测试功能、测试步骤、预期结果和实际结果四个部分。在测试用例时，不仅要测试用例的功能，用户在正常输入条件下系统的运行情况，

还应考虑在黑盒测试和白盒测试下系统的运行情况<sup>[58]</sup>。下面以系统功能模块和辅助系统功能模块设计测试用例，并总结测试结果。

### 6.2.1 全能型供电所管理系统功能模块测试

选取全能型供电所管理系统部分功能模块进行测试，包含工单管理、核算管理、系统综合管理。

#### (1) 全能型供电所管理系统的工单管理模块测试

根据用户在工单管理模块输入相关关键词后，点击“查询”按钮，系统会自动将用户输入的信息与数据库中的信息进行比对，不符合则查询失败，符合则查询成功。工单管理模块测试用例表如表 6-1 所示：

表 6-1 全能型供电所管理系统工单管理模块测试用例表

测试功能	测试步骤	预期结果	实际结果
工单管理	点击“添加”按钮，填写信息内容	填写信息，新增工单	成功
工单管理	点击“添加”按钮，不填写信息内容	提醒未填写	成功
工单管理	点击“处理”按钮	编辑工单、完成工单	成功
工单管理	选中工单，点击“删除”按钮	批量删除和单个删除指定工单	成功
工单管理	点击“终止”按钮	工单终止	成功
工单管理	根据关键词查询工单	根据关键词筛选工单明细	成功
工单管理	输入符合查询条件的关键词	查询到信息	成功
工单管理	输入不符合查询条件的关键词	查不到信息	成功

#### (2) 全能型供电所管理系统的核算管理模块测试

核算管理模块是电费信息进行管理，主要展示电费明细的测试情况。通过点击“欠费/正常”按钮和点击“生成工单”按钮，筛选出“欠费/不欠费”用户信息和将选中某户生成“电费管理”工单，实际结果均为成功。核算管理模块测试用例表如表 6-2 所示：

表 6-2 全能型供电所管理系统核算管理模块测试用例表

测试功能	测试步骤	预期结果	实际结果
电费明细	点击“欠费/正常”按钮	筛选出欠费/不欠费用户信息	成功
电费明细	点击“导出”按钮	导出电费明细表格	成功
电费明细	点击“生成工单”按钮	将选中某户生成“电费管理”工单	成功
电费明细	根据关键词查询电费明细	根据关键词筛选电费明细	成功
电费明细	输入不符合查询条件的关键词	查不到信息	成功

## (3) 全能型供电所管理系统的综合管理模块测试

综合管理模块用于管理所务公开信息和员工信息,主要展示所务基础信息维护和员工信息管理的测试情况。修改所务基础信息后,点击“保存”按钮,信息保存成功,所务公开信息同时改变。通过关键词筛选查询所需的员工信息,点击“添加”和“删除”按钮后,在数据库中同步新增和删除员工信息。系统管理模块测试用例表如表 6-3 所示:

表 6-3 全能型供电所管理系统综合管理模块测试用例表

测试功能	测试步骤	预期结果	实际结果
基础信息维护	修改信息后点击“保存”按钮	信息保存成功,所务公开信息同时改变	成功
员工管理	点击“编辑”按钮	修改员工信息,展示员工信息修改	成功
员工管理	点击“添加”按钮	新增员工信息	成功
员工管理	选中员工信息,点击“删除”按钮	删除员工信息	成功
员工管理	根据关键词查询员工信息	根据关键词筛选员工信息	成功
员工管理	输入不符合查询条件的关键词	查不到信息	成功

### 6.2.2 辅助全能型供电所管理系统功能模块测试

系统辅助功能模块主要包括用户登录、注册和退出三个功能。在用户登录或者注册时，若未输入用户名、密码和验证码，系统会弹出提示信息。当用户登录时，输入用户名、密码和验证码后，系统会自动比对用户输入的信息与数据库中存储的数据。如果信息一致，则登录操作成功；如果不一致，系统将自动弹出相应的提示信息，告知用户登录失败。系统辅助功能模块测试用例表如表 6-4 所示：

表 6-4 全能型供电所管理系统辅助功能模块测试用例表

测试功能	测试步骤	预期结果	实际结果
用户登录	未输入用户名和密码	提示“请输入用户名”和“请输入密码”	成功
用户登录	未输入验证码	提示“请输入验证码”	成功
用户登录	输入正确用户名和密码	登录系统主页面，并提示“登录成功”	成功
用户登录	输入未注册的用户名	提示“用户不存在”	成功
用户登录	输入错误的密码	提示“用户密码错误”	成功
用户登录	输入错误的验证码	提示“验证码错误”	成功
用户注册	输入已注册的用户名	提示“用户已经存在，请重新输入”	成功
用户注册	未输入账号和密码	提示“请输入用户名”和“请输入密码”	成功
修改用户信息	登录成功后，点击“修改个人信息”菜单	成功修改个人信息	成功
强制退出登录	超过 1 小时，用户是否会退出登录	刷新后，用户退出登录	成功
手动退出登录	登录成功后，点击“退出登录”菜单	用户退出登录	成功

### 6.3 单元测试

在完成系统功能模块测试的过程中，本小节使用白盒测试的单元测试全面测试系统代码中的最小可测单元。

通过提高测试效率和准确性，在执行单元测试过程中需要充分利用辅助工具。本系统采用集成开发环境 IDEA 中内嵌的 Junit 测试框架作为主要测试工具，针

对本系统中的 Aop、Listener、Utils、Service 四个关键单元设计测试用例，对系统代码内定义的函数与方法进行全面的验证，四个单元测试覆盖率统计如表 6-5 所示：

表 6-5 单元测试覆盖率统计表

测试单元	类个数	方法数	行数	覆盖率
Aop	1	2	109	95%
Listener	6	14	491	92%
Utils	4	8	171	94%
Service	19	99	1724	92%
合计	30	123	2495	93%

通过对上述四个单元的代码执行覆盖率结果进行分析，尽管存在局部未覆盖的情况，但测试的广度与深度总体上满足了系统预定的标准。

## 6.4 非功能性测试

对于系统的性能、安全性、兼容性、易用性等方面进行测试可以采用非功能性测试，通过非功能性需求测试，保证系统可以稳定地提供相关服务，给用户良好的用户体验<sup>[59]</sup>。

### 6.4.1 性能测试

性能测试是通过性能测试工具模拟各种性能负载场景下系统运行情况。性能测试的目的是测试系统在不同的条件下各方面的性能指标，包括并发测试和压力测试<sup>[60]</sup>。

压力测试又包含系统吞吐量、系统响应时间和系统在不同时间的活动用户数量展示，验证系统的稳定性。本文使用 JMeter 压力测试工具对系统登录功能和其他页面访问进行并发测试和压力测试。首先创建线程组，填写相关的设置参数，添加 HTTP 请求，配置 HTTP 信息，测试执行过程即可启动。其次需在配置中设定线程数，这代表模拟的用户数量。接着通过监控和分析系统在不同压力条件下的响应时间、最短时间、最长时间、吞吐量以及其他异常情况，能够全面评估系统的性能表现。最后添加监听器，通过聚合报告查看并分析测试结果。



### (1) 并发测试

使用 JMeter 在 30 秒内启动完 600 个线程，相当于每秒启动 20 个进程。进程循环 10 次，总计 6000 次。测试出现的错误请求数量百分比为 0%，系统的平均响应时间为 300ms 以内，服务器每秒处理的请求事务数为 50 个以内，说明系统中不存在可能隐藏的并发访问的问题。具体情况如图 6-1 所示：

Label	# 样本	平均值	最小值	最大值	标准偏差	异常 %	吞吐量	接收 KB...	发送 KB...	平均字节数
HTTP请求	6000	270	7	1157	173.04	0.00%	45.1/sec	26.85	5.11	610.0
总体	6000	270	7	1157	173.04	0.00%	45.1/sec	26.85	5.11	610.0

图 6-1 系统并发测试图

### (2) 系统吞吐量

使用 JMeter 在 5 分钟内，开启 300 个线程，请求共 30000 次，对系统的系统吞吐量进行测试，结果是系统每秒钟能处理的用户请求数量在 300 个左右，请求错误率在 0.14%，说明系统处理能力极好。具体情况如图 6-2 所示：

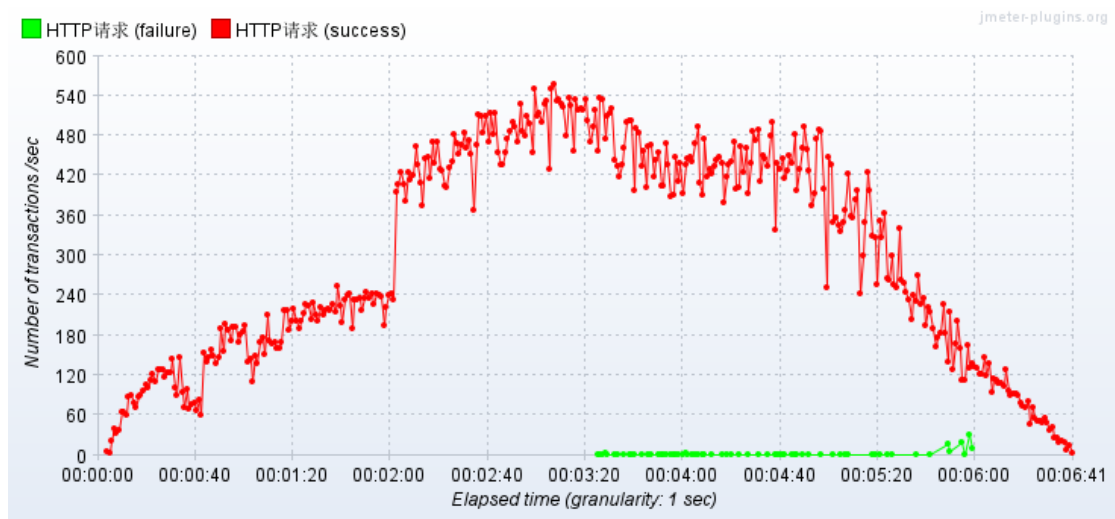


图 6-2 系统吞吐量图

### (3) 系统响应时间

使用 JMeter 在 5 分钟内，开启 300 个线程，请求共 30000 次，对系统的响应时间进行测试，响应时间全都在 735ms 以内，说明系统的响应速度非常快。具体情况如图 6-3 所示：

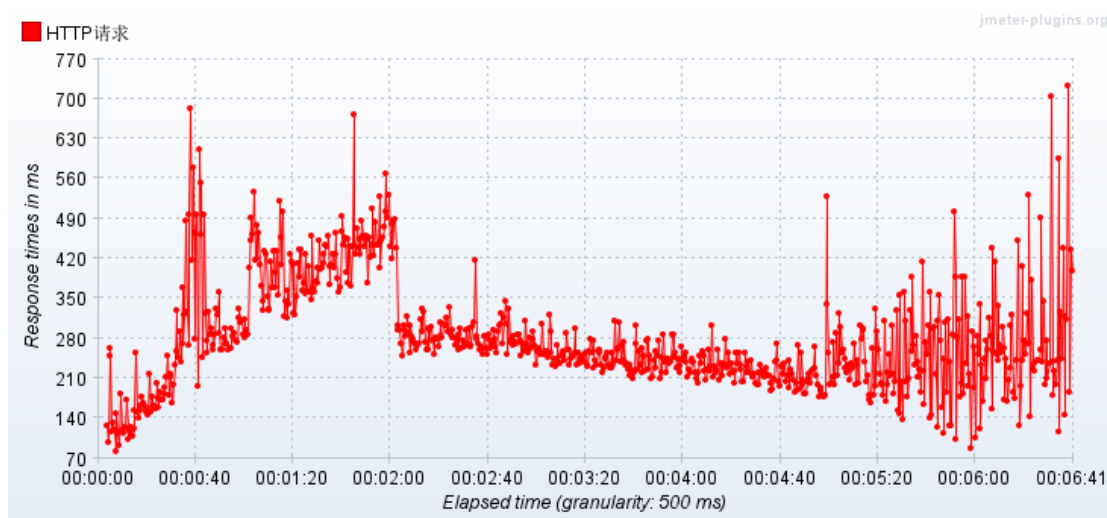


图 6-3 系统响应时间图

#### (4) 系统在不同时间的活动用户数量展示

使用 JMeter 在 10 分钟内, 开启 100 个线程, 请求共 60000 次, 对系统不同时间活动用户数量展示, 结果是在 2 分 59 秒左右, 活动用户数量达到最大为 140, 从 2 分 59 秒至 6 分 41 秒, 活动用户数量逐渐减少, 直至为 0, 在测试过程中没有发生异常或崩溃情况, 说明系统的承受压力能力比较强、性能优秀, 具体情况如图 6-4 所示:

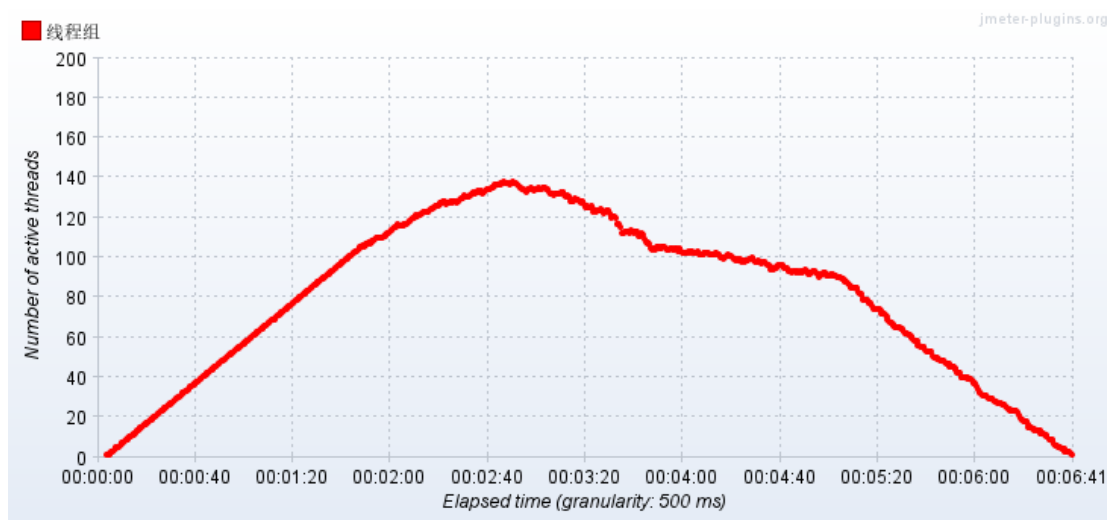


图 6-4 系统响应时间分布图

6.4.2 其他非功能性测试

系统在测试过程中应该进行功能性测试、单元测试、性能测试，同时也需要对系统其他非功能性展开测试。通过对系统安全性、系统兼容性和系统易用性三个方面展开测试用例测试，预期结果显示系统安全性高、兼容性强、简单易用，实际结果均成功。系统其他非功能性测试用例表如表 6-6 所示：

表 6-6 全能型供电所管理系统的其他非功能性测试用例表

测试功能	测试步骤	预期结果	实际结果
系统安全性	系统用户密码是否加密	系统用户密码已加密	成功
系统兼容性	用不同浏览器打开系统，并进行模块测试	测试三个浏览器兼容	成功
系统易用性	站在用户角度体验系统使用感	操作简单、界面简洁	成功

6.5 系统测试结果分析

在系统测试中，对系统的功能性和非功能性展开测试。除了考虑系统正常情况下的测试用例，还需要考虑黑盒测试、白盒测试、并发测试、压力测试等其他测试。经过这些不同的测试，本系统基本实现了页面、菜单和功能模块，系统符合实际需求。

6.6 本章小结

本章对整个系统的功能性和非功能性进行了测试，测试结果符合预期，整体没有问题。

## 第七章 总结与展望

### 7.1 总结

随着时代的日益发展，技术的不断进步，系统开发逐步走向智能时代。在日常工作中离不开各种系统的支撑，一个系统的好坏影响着我们的日常工作，因此系统在我们的工作中扮演着重要的角色。我们人类也是通过不断学习，增强自身业务能力和学习能力，更好地适应工作和生活。系统更应该如此，系统也需要更新迭代，适应新时代发展和高质量工作完成的需要。因此，本人结合自身工作，设计并实现出全能型供电所管理系统。

本文的研究主要包括以下几个方面：

(1) 本文分析全能型供电所管理系统的研究背景和意义，以及目前供电所管理系统国内外发展及研究现状。

(2) 对系统实现运用到的一些框架和技术做了详细的介绍，系统使用 SpringBoot 和 Vue 框架，前后端分离，真正实现前后端解耦，加快整体响应速度。将 Vue 框架和 Element-UI 相结合，简洁用户界面，提供良好的人机交互体验。采用 AOP 技术，实现日志拦截。采用 TK-Mapper 框架，使用通用方法和自定义通用方法进行表单查询。通过 ECharts 库绘制分布图，提供可视化数据展示。采用 Axios 库执行 HTTP 请求。通过 Lombok 库编写注解简化编程，提高开发效率。采用 Shiro 安全机制，严格控制权限访问及系统安全。使用 MySQL 和 Redis 数据库交互操作的形式，加快访问速度。实现基于 TopN 和权重乘积之和的自动派单算法，代替了人工繁琐的派单步骤，避免了人工录入出错，有效提升了数据质量及派单准确性。基于最小二乘法的用电量预测算法，能够提前预知客户用电量，及时告知客户电力峰值，帮助用户减少用电成本。

(3) 对全能型供电所管理系统进行全面的需求分析，包含了总体需求分析、用户需求分析、功能性需求和非功能性需求分析。并对系统各个模块进行了详细的划分，确定系统的主要功能。

(4) 对系统进行详细设计, 包括系统总体架构设计、功能模块设计和数据库设计, 对关键功能模块进行算法设计。

(5) 设计并实现全能型供电所管理系统, 对系统各个模块的实现进行详细描述, 并对关键功能界面进行展示, 对关键功能的部分代码进行描述。

(6) 对系统环境、功能性和非功能性模块展开测试。

## 7.2 展望

全能型供电所管理系统各个功能模块和框架大体上已经完成, 但是基于现实需求, 仍有拓展的空间:

- (1) 将接单功能拓展到手机端, 易于随时随地都能够接单。
- (2) 完善其他供电所相关功能模块和数据。
- (3) 系统的功能还需要进一步扩充。

## 参考文献

- [1] 王宇哲. 电力大用户电费回收风险预警管理系统研究及软件开发[D]. 西华大学, 2016.
- [2] 张懿富. 乡镇供电所一体化系统关键技术探索[J]. 科技经济导刊, 2020, (28):52-57.
- [3] 汪一浩, 周开保. "三集五大"体系下乡镇供电所管理模式探讨[J]. 中国电力企业管理, 2014, (24):88-90.
- [4] 谢晨, 赵广杰. 服务乡村振兴战略的乡镇供电所布局优化创新[J]. 管理学家, 2022, (1):94-96.
- [5] 本刊编辑部. "全能型"乡镇供电所建设再提升[J]. 农电管理, 2020, (01):1.
- [6] 徐华, 吕顺利, 王路畅等. 营配融合移动应用框架[J]. 中国设备工程, 2019, (04):231-232.
- [7] 冯晋, 曾东. 供电所优化提升助力建设服务新格局[J]. 2022, (29):80-81.
- [8] 魏文江. 浅议我国当前电力市场化改革[J]. 时代经贸, 2013(20):89-89.
- [9] 宁瑞琪. 对"两改一同价"决策的理解和分析[J]. 电力技术经济, 2003(4):13-17.
- [10] 厂网分开 打破垄断 中国国家电力公司拆分重组成 11 家[J]. 继电器, 2003(01):16.
- [11] 汤文全. 真抓实干开拓创新全面提升公司创一流同业对标水平[J]. 湖北电业, 2007(2):8-10.
- [12] 王宏, 崔勇. 电力企业同业对标管理的研究[J]. 价值工程, 2014(2):172-172, 173.
- [13] 崔梅雪. 同业对标在供电所管理中的应用研究[J]. 百科论坛电子杂志, 2020(14):1672.
- [14] 李林威, 赵喆, 刘帮成. "放管服"背景下我国电力体制改革成效评估及优化路径研究——基于重庆市 1486 家用电企业电力面板数据分析[J]. 公共行政评论, 2021, 14(5):140-158.

- [15]徐杰彦, 卢悦, 曲弘, 等. 新电改背景下我国电网公司售电服务竞争力分析[J]. 电力需求侧管理, 2020, 22(4):89-93, 100.
- [16]杨歌. 国家电网公司:5 年内建成国际领先的能源互联网企业[J]. 电力系统装备, 2021(3):14-15.
- [17]蔡雨楠. 新形势下电网企业数字化转型实施策略研究[J]. 经济师, 2019(05):271-272+275.
- [18]刘早. 国家电网公司第四届职工代表大会第四次会议暨 2024 年工作会议召开[N]. 国家电网报, 2024-01-13(001).
- [19]龙兰英, 施其祥, 张学超. 农村供电所管理模式研究[J]. 农电管理, 2021(12):70-71.
- [20]邢宪蕊, 陈泓安. 日庄供电所“:党建+新绩效”助力管理转型[J]. 中国电力企业管理, 2023, (29):20-21.
- [21]李献才, 张灯耀, 黄立群等. 构建“中央厨房式”乡镇供电所管理体系[J]. 中国电力企业管理, 2023, (29):16-17.
- [22]吴祖培, 丘晓峰, 李献才等. 古田供电所:“机器人”助力线损管理增效[J]. 中国电力企业管理, 2023(29):46-47.
- [23]李刚. 供电所网格化管理的探索与思考[J]. 农村电工, 2023, 31(10):19.
- [24]Muttaqi K M, Ledwich G, Improvement of power supply to rural customers through grid-interactive distributed generation[J]. International Journal of Global Energy, 2006, 26(3/4):341-360.
- [25]Singh S N, John P, Prabhakar N. Technical Feasibility of Photovoltaic Fuel Cell:A Model of Green Home Power Supply for Rural India[J]. International Journal of Emerging Trends & Technology in Computer Science, 2012, 1(2):216-202.
- [26]Harish S M. Morgan G M . Subrahmanian E. When does unreliable grid supply become unacceptable policy? Costs of power supply and outages in rural Indian[J]. Energy Policy, 2014, 68:158-169.
- [27]李洁. 主流 JavaScript 框架——AngularJS、React 和 Vue 使用体会[J]. 电脑迷,

- 2019, 000(1):77.
- [28]陈润毅. 基于物联网的家庭园艺系统的设计与实现[D]. 厦门理工学院, 2021.
- [29]王杰, 薛博宇, 徐云华等. 基于 C/S 架构和物联网技术的可移动医疗设备共享调配系统的设计与应用[J]. 中国医疗设备, 2024, 39(01):85-91.
- [30]满春梅. 基于 B/S 架构的建筑工程项目管理系统设计与功能实现[J]. 科技创新, 2023, (18):180-183.
- [31]边蓓蓓, 于萍. MVC 模式在 Web 中的应用研究[J]. 数字技术与应用, 2015(10):98.
- [32]危炜. 基于 SpringBoot 和 Vue 框架的刹车片工厂流程再造系统的设计与实现[D]. 浙江理工大学, 2021.
- [33]翟剑锟. Spring 框架技术分析及应用研究[D]. 中国科学院大学, 2013.
- [34]Pivotal. Spring Boot Reference Guide1. 5. 3. RELEASE[OL]. 2017.
- [35]Walls C. Spring Boot in Action[M]. Manning Publications Co. 2016.
- [36]陈涛, 叶荣华. 基于 SpringBoot 和 MongoDB 的数据持久化框架研究[J]. 电脑与电信, 2016(1):71-74.
- [37]彭炳江. 基于政府人才云平台的申报审批系统研究与实现[D]. 贵州大学, 2021.
- [38]朱二华. 基于 Vue. js 的 Web 前端应用研究[J]. 微计算机信息, 2017(20):119-121.
- [39]Li D, Mei H, Shen Y, et al. echarts:a declarative framework for rapid construction of web-based visualization[J]. Vis. Informatics, 2018, 2(2):136–146.
- [40]叶济凡. 基于 SpringBoot 与 Vue 框架的中文社科论文分析系统的设计与实现[D]. 南京大学, 2020.
- [41]骆文亮. 绘图插件 Highcharts 浅析[J]. 科技视界, 2014(12):184.
- [42]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.
- [43]徐孝成. 基于 Shiro 的 Web 应用安全框架的设计与实现[J]. 电脑知识与技术,



- 2015(16):93-95.
- [44]刘涛. 基于 SpringBoot 的银行二维码交易处理系统设计与实现[D]. 西安电子科技大学, 2020.
- [45]顾少伟, 井波. Redis 在软件项目中的应用[J]. 电脑编程技巧与维护, 2023, (11):16-19+32.
- [46]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.
- [47]关沫, 魏碧晴. 基于 Flink 框架的 TopN 堆排序优化算法[J]. 信息技术与网络安全, 2020, 39(02):23-26.
- [48]刘丰威, 董茵, 张卓. 基于线性拟合算法的用电量异常阈值的计算[J]. 中国高新科技, 2022(20):5-7.
- [49]单宇, 许晖, 周连喜, 周琪. 数智赋能:危机情境下组织韧性如何形成? ——基于林清轩转危为机的探索性案例研究 [J]. 管理世界, 2021, 37(03):84-104+7.
- [50]胡腾耀. B 供电公司电费回收风险管理研究[J]. 质量与市场, 2023(02):34-36.
- [51]张莉. 基于 Web 服务的分布式信息共享系统的研究与实现[D]. 西安电子科技大学, 2014.
- [52]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.
- [53]应进平. 浅谈 C 语言程序设计中算法设计的作用[J]. 科技情报开发与经济, 2006, 16(24):261-262.
- [54]刘敏. 基于 SpringBoot 框架社交网络平台的设计与实现[D]. 湖南大学, 2018.
- [55]迟承哲. 供电企业物资管理系统设计与实现[D]. 电子科技大学, 2010.
- [56]Esnaashari Mehdi, Damia Amir Hossein. Automation of software test data generation using genetic algorithm and reinforcement learning[J]. Expert Systems With Applications, 2021, 183.
- [57]刘肖. 基于 JSP+MySQL 的供电管理系统的设计与实现[D]. 电子科技大学,

2021.

- [58]FRANCESCOP, GAIA R, MASSIMOD I, et al. A Cooperation Project in Lesotho:Renewable Energy Potential Maps Embedded in a WebGIS Tool [J]. Sustainability, 2021, 13(18).
- [59]潘瑜. 移动报考系统的设计与实现[D]. 华南理工大学, 2017.
- [60]马玉洁. 基于云办公的技术经济管理业务系统设计与实现[D]. 电子科技大学, 2019.