



山东大学
SHANDONG UNIVERSITY

Shandong University

School of Computer Science and Technology

软件需求规格说明

Software Requirements Specification (SRS)

Project : 人事管理系统

Team Member : 宋天恩、张疆龙、陈由亮、隋沅松、韩鲁

目录

软件需求规格说明(SRS).....	错误!未定义书签。
1 范围	4
1.1 标识.....	4
1.2 系统概述.....	4
1.3 文档概述.....	5
1.4 基线.....	5
2 引用文件.....	6
3 需求	6
3.1 所需的状态和方式.....	6
3.2 需求概述.....	6
3.2.1 目标.....	6
3.2.2 运行环境.....	8
3.2.3 用户的特点.....	8
3.2.4 关键点.....	8
3.2.5 约束条件.....	9
3.3 需求规格.....	9
3.3.1 软件系统总体功能/对象结构.....	9
3.3.2 软件子系统功能/对象结构.....	13
3.3.3 描述约定.....	14
3.4CSCI 能力需求.....	14
3.5CSCI 外部接口需求.....	19
3.5.1 接口标识和接口图.....	20
3.6CSCI 内部接口需求.....	22
3.7CSCI 内部数据需求.....	22
3.8 适应性需求.....	23
3.9 保密性需求.....	24
3.10 保密性和私密性需求.....	25
3.11CSCI 环境需求.....	26
3.12 计算机资源需求.....	26
3.12.1 计算机硬件需求.....	26
3.12.2 计算机硬件资源利用需求.....	26
3.12.3 计算机软件需求.....	27
3.12.4 计算机通信需求.....	27
3.13 软件质量因素.....	27
3.14 设计和实现的约束.....	27
3.15 数据.....	27
3.16 操作.....	28
3.17 故障处理.....	28
3.18 算法说明.....	28

3.19 有关人员需求.....	28
3.20 有关培训需求.....	29
3.21 有关后勤需求.....	29
3.22 其他需求.....	29
3.23 包装需求.....	29
3.24 需求的优先次序和关键程度.....	29
4 合格性规定.....	30
5 需求可追踪性.....	31
6 尚未解决的问题.....	32

1 范围

1.1 标识

标识号: Human Resource Management System-version1.0

标题: 人事资源管理系统软件需求分析说明

缩略词语: HRMS-V1.0

版本号 Alpha 版

发行号: 1.1.1.240321_alpha

1.2 系统概述

1.2.1: 人事资源管理系统的项目用途

人事资源管理系统是一款专门为企业设计的人力资源管理软件,旨在通过集中、系统的方式管理企业员工的各类信息,包括但不限于员工档案、薪资、考勤、培训、绩效等。该系统通过自动化和智能化的手段,提高人事管理效率,优化人力资源配置,为企业决策提供数据支持,从而提升企业的整体运营效率和员工满意度。

1.2.2: 本系统所具有的特性:

- a. 集中管理: 实现员工信息的集中存储和管理,方便随时查询和更新。
- b. 自动化处理: 支持人事流程的自动化处理,如入职、离职、转岗等。
- c. 数据分析: 提供丰富的数据分析和报表功能,帮助企业做出更明智的决策。
- d. 多角色协同: 支持多部门、多角色的协同工作,确保信息的及时共享和沟通。
- e. 安全可靠: 系统遵循严格的数据保护法规,确保员工信息的安全性和隐私性。

1.2.3: 项目开发、运行和维护历史概述

该人事资源管理系统自立项以来,经历了需求调研、系统设计、开发实施、测试验收等多个阶段,期间进行了多次版本更新和功能优化,以满足企业不断变化的需求。同时,为了确保系统的持续稳定运行,接下来我们准备建立了专业的维护团队,负责系统的日常维护和故障处理。

1.2.4: 项目投资方、需方、用户、开发方和支持机构标识

- a. 投资方: 暂无
- b. 需方: 有人事资源管理需求的中大型企业
- c. 用户: 相关企业人事部门员工及高层管理人员
- d. 开发方: 本软件工程课设小组
- e. 支持机构: 暂无

1.2.5: 当前和计划的运行现场

当前本系统还在研发阶段,尚不能投入正常使用。

我们期待在将来投入应用之后,该系统可以在多家企业成功应用,并收到大家的好评。

未来，我们计划将系统推广至更多行业和企业，以满足更多用户的需求。

1.2.6: 其他有关的文档

- a. 需求规格说明书：详细描述了用户对系统的功能和性能需求，以及系统的架构设计、数据库设计、界面设计等。
- b. 测试报告：记录了系统测试的过程和结果，包括功能测试、性能测试等。
- c. 用户手册：为用户提供了系统的操作指南和常见问题解答。
- d. 维护手册：为维护人员提供了系统的维护指南和故障处理流程。

1.3 文档概述

1.3.1 文档的用途

本《人事资源管理系统需求规格说明（SRS）》旨在详细阐述人事资源管理系统的各项功能需求、性能需求、安全性需求以及系统的非功能性需求，为开发团队提供明确的开发方向和标准，同时也为项目管理者、测试团队以及其他利益相关者提供统一的参考依据。通过本文档的详细描述，确保系统的设计与开发能够准确满足用户的实际需求，为企业的人力资源管理提供有力的信息化支持。

1.3.2 文档内容:

- (1) 范围：标识、系统概述、文档概述、基线
- (2) 引用文件
- (3) 需求：需求概述、需求规格、CSCI 能力概述。接口需求等。
- (4) 合格规定性
- (5) 需求可追踪性

1.4 基线

(1) 系统结构设计：设计系统的整体结构，包括前端界面、后端服务、数据库等组件之间的交互关系。

(2) 模块划分：将系统功能划分为模块，定义各模块的职责和接口，确保模块之间的高内聚低耦合。

(3) 技术选型：选择合适的开发语言、数据库、框架和工具，以满足系统的性能、安全性、可维护性和扩展性要求。考虑到系统的实际需求和未来发展方向，选择适合的技术栈。

(4) 安全性设计：设计系统的安全策略，包括用户身份认证、权限管理、数据加密、防止数据泄露等措施，确保系统数据的安全性和隐私保护。

(5) 用户体验设计：设计直观友好的用户界面，提供良好的用户体验，使用户能够轻松使用系统进行操作。

考虑不同用户角色的需求，定制化界面以满足其特定需求。

(6) 性能优化：设计高效的系统架构和算法，以保证系统的响应速度和稳定性。

考虑系统的扩展性和并发性，以应对未来可能的用户增长和业务扩展。

(7) 可维护性设计：设计清晰的代码结构和文档，便于团队成员理解和维护系统。

考虑到系统的可扩展性和升级性，确保系统能够方便地进行功能扩展和升级。

2 引用文件

编号	标题	修订版本	日期
1	《掌握需求过程》	V3.0	2014-01-01
2	《软件工程导论》	V4.0	2010-05

3 需求

3.1 所需的状态和方式

员工和管理员根据各自实际身份登录人事管理系统，登陆成功则启动相应的管理界面，提供相应的权限，实现各项功能。

3.2 需求概述

3.2.1 目标

a. 本系统的开发意图、应用目标及作用范围(现有产品存在的问题和建议产品所要解决的问题)。

开发意图：

人事管理系统的开发意图在于通过信息技术手段,提高企业人力资源管理的效率和准确性。通过构建一个集成化、自动化的管理平台,实现员工信息的统一管理和维护,简化人力资源管理流程,降低管理成本,并为企业提供数据支持和决策依据。

应用目标：

通过人事管理系统,企业可以将员工的个人信息、合同、薪资、考勤等各类数据集中存储和管理,确保数据的准确性和一致性。解决当前许多公司员工信息、薪资福利、考勤等数据可能分散在不同的系统或文件中,导致数据不一致和查询困难。此系统能够实现员工入职、转正、调岗、离职等流程的自动化处理,减少人工操作,提高工作效率,应对现有公司工作效率低的问题。此外,此系统还能为员工提供自助查询和更新个人信息的功能,增强员工满意度和参与度。

作用范围：

人事管理系统的作用范围涵盖企业人力资源管理的各个方面,包括但不限于员工信息管理、薪资福利管理、考勤管理、招聘管理、培训管理以及绩效考核等。系统适用于各类型企业,特别是那些需要高效、精准管理人力资源的企业。

b. 本系统的主要功能、处理流程、数据流程及简要说明。

主要功能：

- (1) 员工各种信息的输入，包括员工的基本信息、学历信息、婚姻状况信息、职称等。
- (2) 员工各种信息的修改；
- (3) 对于转出、辞职、辞退、退休员工信息的做相应标记，默认显示在职员工信息；
- (4) 工资、奖金、扣款细节可以自定义，员工可以网上查看个人的工资条
- (5) 按照一定的条件，查询、统计符合条件的员工信息；至少应该包括每个员工详细信息的查询、按婚姻状况查询、按学历查询、按工作岗位查询等，至少应该包括按学历、婚姻状况、岗位、参加工作时间、工资等统计各自的员工信息；对查询、统计的结果打印输出。

处理流程：

员工入职流程：包括招聘需求确认、发布招聘信息、简历筛选、面试、录用等。

员工离职流程：包括离职申请、离职审批、交接工作、办理离职手续等。

薪资发放流程：包括薪资核算、薪资发放、薪资调整等。

绩效考核流程：包括设定绩效目标、员工自评、主管评定、绩效面谈、绩效考核结果确认等。

数据流程：

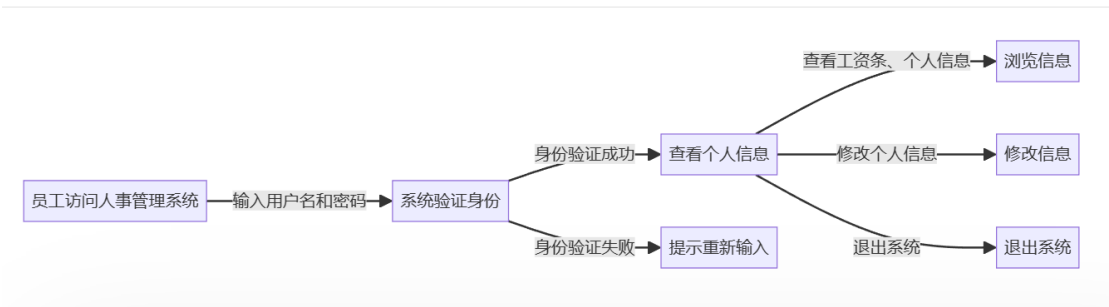
员工信息流程：员工信息的录入、修改、查询和删除过程。

薪资数据流程：薪资核算、发放、调整的数据流转过程。

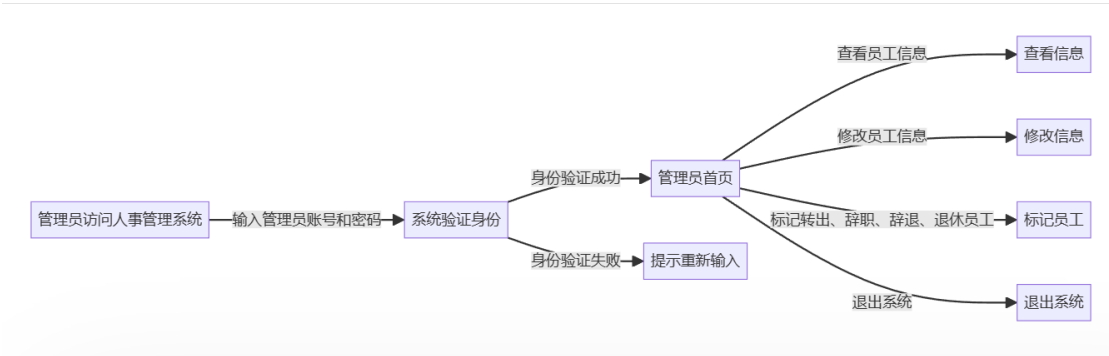
考勤数据流程：考勤记录、请假、加班等数据的采集和处理流程。

培训数据流程：培训计划、培训记录、培训效果评估等数据的流转过程。

员工：



管理员：



3.2.2 运行环境

硬件环境:

处理机: AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz

RAM: 16GB

系统: 64 位操作系统, 基于 x64 的处理器

开发工具: 项目将使用 Vue 3 作为主要的前端框架, 配合相应的开发工具如 Vue CLI、Visual Studio Code 等。

* JDK : Oracle JDK 1.8

* Mysql: 5.7

* Maven: 3.5.2

* Redis: 2.4.5

服务器环境: 项目可能需要配置本地或云端的服务器环境, 以支持后端数据处理和数据库存储。

网络环境: 项目应在稳定的网络环境下进行, 以确保团队成员之间的远程协作和数据同步。

3.2.3 用户的特点

人事管理系统的用户主要以员工较多, 员工信息较杂的企业为主。它们主要特点为:

1. 员工以员工较多, 员工信息较杂。
2. 企业用户倾向于通过人事管理系统实现人力资源管理的标准化和规范化。
3. 企业重视数据分析, 希望通过人事管理系统收集、整理和分析员工数据, 为管理层提供决策支持。
4. 企业用户注重工作的效率性。希望通过系统的自动化功能, 减少繁琐的手工操作, 从而加快流程处理速度, 降低出错率。

3.2.4 关键点

关键功能:

员工信息管理: 包括员工档案管理、个人信息、工作经历、培训记录等的管理和维护。

薪酬福利管理: 薪资核算、福利管理、绩效考核等与员工薪酬相关的关键功能。

考勤管理: 记录员工考勤情况、请假管理、加班管理等关键考勤功能。

绩效管理: 设定绩效目标、绩效评估、绩效报告生成等关键绩效管理功能。

关键算法:

薪资核算算法: 计算员工薪资的算法, 包括基本工资、津贴、扣款等的计算规则。

绩效评估算法: 评估员工绩效的算法, 包括绩效指标的权重分配、评分规则等。

招聘筛选算法: 简历筛选和面试评估的算法, 用于自动化筛选和评估候选人。

涉及的关键技术:

数据库技术: 用于存储和管理员工信息、薪资数据、考勤记录等相关数据的数据库技术。

Web 开发技术: 用于实现系统的用户界面和交互功能的 Web 开发技术, 如 HTML、CSS、

Vue3 等。

数据安全技术：包括用户身份验证、数据加密、访问控制等技术，确保系统数据的安全性。

集成技术：用于与其他系统接口和数据交互的集成技术，确保系统与外部系统的无缝连接。

3.2.5 约束条件

经费限制：开发人员需要在预算范围内完成系统的开发工作，经费限制可能影响到开发周期、开发人员数量、使用的软硬件设备等方面。

技术限制：由于团队成员可能缺乏丰富的项目经验，对于一些复杂的技术问题可能需要额外的学习和时间来解决。

时间限制：作为课程项目，项目的时间周期相对较短，可能需要在有限的时间内完成较多的功能开发。

资源限制：项目可能受到学校或课程组织方提供的资源限制，如服务器配置、数据库容量等。

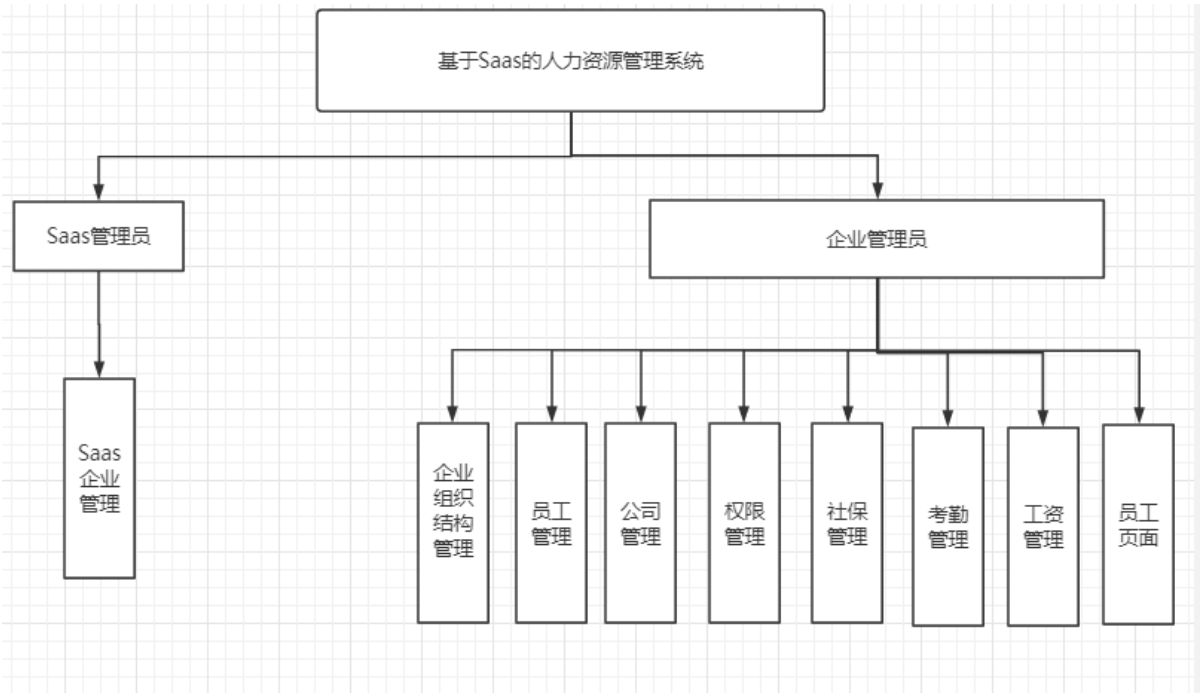
政治、社会、文化、法律等方面的限制：开发人员需要遵守相关政策、法规和标准，确保系统的合法性和安全性。此外，还需要考虑不同文化背景、语言、习惯等因素对系统设计和开发的影响。

3.3 需求规格

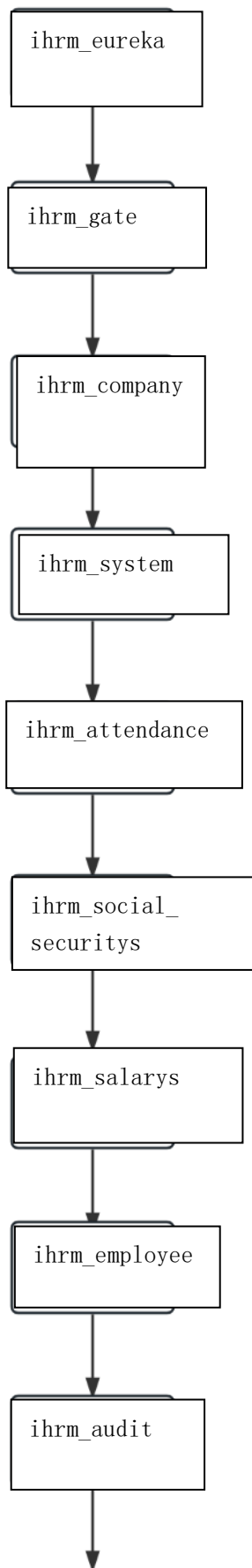
3.3.1 软件系统总体功能/对象结构

对软件系统总体功能/对象结构进行描述，包括结构图、流程图或对象图。

结构图：



流程图:



启动顺序原因：
因为所有模块都注册到了 Eureka 中，故最先启动 Eureka 模块
因为在 gate 模块后面都是业务模块，所以网关模块放在第二位启动
因为 system 模块通过 feign 调用了 company 模块的接口，所以 company 要在 system 模块之前启动
因为 salarys 模块通过 feign 调用了 attendance 和 social_securitys 模块的接口，所以 salarys 在其之后启动
后面两个顺序可以打乱
服务调用了其他服务的接口，最好是先去启动被调用的模块

对象图：

员工
ID
姓名
部门
职位
入职日期
薪资

薪酬
基本工资
奖金
扣款
总薪资

考勤对象
签到时间
签退时间
迟到早退
请假
加班

绩效对象
绩效目标
得分和评估

3.3.2 软件子系统功能/对象结构

对每个主要子系统的基本功能模块/对象进行描述，包括结构图、流程图或对象图。



1. 考勤模块：考勤模块是人事资源管理系统的核心部分，主要用于记录和管理员工的出勤情况。该模块可以实时追踪员工的上下班时间、请假、加班、迟到、早退等考勤信息，并通过自动化的数据处理和分析，为管理者提供考勤统计和报表。此外，考勤模块还可以与其他模块（如薪资模块）进行集成，确保考勤数据能够准确、及时地应用于薪酬计算和绩效评估。
2. 流程模块：流程模块主要负责管理企业内部的各项流程，如招聘、入职、转正、调岗、离职等员工生命周期流程。通过流程模块，企业可以自定义和优化流程，提高流程执行的效率和准确性。同时，该模块还可以提供流程监控和审批功能，确保流程的合规性和透明度。
3. 企业模块：企业模块通常用于管理企业的基本信息，如组织架构、部门设置、岗位职责等。该模块可以帮助企业建立清晰的组织架构，明确各部门和岗位的职责和权限，为企业的运营和管理提供有力的支持。
4. 员工模块：员工模块是人事资源管理系统中用于管理员工信息的核心模块。它涵盖了员工的基本信息、教育背景、工作经历、技能特长等各个方面。通过员工模块，企业可以全面了解员工的情况，为招聘、培训、晋升等决策提供数据支持。
5. Eureka 模块：Eureka 是 Netflix 的一个子模块，主要用于服务注册和发现。在人事资源管理系统中，Eureka 模块可以应用于微服务架构中，实现服务之间的自动发现和注册。通过 Eureka 模块，系统可以更加灵活地扩展和部署，提高系统的可用性和稳定性。
6. 网关模块：网关模块通常用于处理系统的请求和响应，实现内外网之间的隔离和安全控制。在人事资源管理系统中，网关模块可以确保系统的安全性和稳定性，防止非法访问和恶意攻击。同时，该模块还可以对请求进行过滤和路由，提高系统的响应速度和处理效率。
7. 薪资模块：薪资模块是人事资源管理系统中的重要组成部分，主要用于计算和管理员工的薪酬。该模块可以根据员工的考勤、绩效、职位等因素，自动计算员工的工资和奖金，并生成相应的工资单和报表。同时，薪资模块还可以提供薪资调整、薪资查询等功能，方便员工和管理者随时了解薪资情况。
8. 社保模块：社保模块主要用于管理员工的社保缴纳情况。该模块可以自动计算员工的社保缴纳金额，并与相关部门进行对接，确保社保缴纳的准确性和及时性。此外，社保模块还可以提供社保查询和报表功能，方便员工和管理者随时了解社保缴纳情况。

9. 公共模块：主要放一些公共的工具类、常量类和一些 service

10. 公共实体模块：主要放每个模块的 request 和 response。

3.3.3 描述约定

数学符号：

+ 和 -：用于表示员工的入职和离职状态或过程。例如，“+”可能表示员工入职，而“-”表示员工离职。

✓ 和 ✗ 或 √ 和 ×：用于表示员工某项任务或培训的完成情况。例如，✓ 或 √ 可能表示任务已完成，而 ✗ 或 × 表示任务未完成。

T 和 F 或 1 和 0：用于表示布尔值，如 T 或 1 代表“是”或“真”，而 F 或 0 代表“否”或“假”。

M, F, ♂, ♀：用于表示员工的性别，其中 M 和 ♂ 代表男性，F 和 ♀ 代表女性。

度量单位：

年、月、日：用于表示员工的入职日期、离职日期、生日等与时间相关的数据。

元、千元、万元：用于表示薪资、奖金等货币相关的数据。

其他约定：

员工编号：唯一标识每个员工的数字或字母组合，确保员工信息的准确性和可追溯性。

职位名称：标准化和统一的职位命名规则，便于查询、统计和分析。

部门划分：根据公司业务和组织结构，合理划分部门，便于管理和协作

3.4 CSCI 能力需求

a. 说明

描述此功能要达到的目标、所采用的方法和技术，还应清楚说明功能意图的由来和背景。

b. 输入

包括：

1) 详细描述该功能的所有输入数据，如：输入源、数量、度量单位、时间设定和有效输入范围等。

2) 指明引用的接口说明或接口控制文件的参考资料。

c. 处理

定义对输入数据、中间参数进行处理以获得预期输出结果的全部操作。包括：

1) 输入数据的有效性检查。

2) 操作的顺序，包括事件的时间设定。

3) 异常情况的响应，例如，溢出、通信故障、错误处理等。

4) 受操作影响的参数。

5) 用于把输入转换成相应输出的方法。

6) 输出数据的有效性检查。

d. 输出

1) 详细说明该功能的所有输出数据，例如，输出目的地、数量、度量单位、时间关系、有效输出范围、非法值的处理、出错信息等。

2) 有关接口说明或接口控制文件的参考资料。

3.4.1 数据处理能力：

需求：CSCI 应具备高效、准确的数据处理能力，能够处理大量的员工数据，包括但不限于个人信息、考勤记录、薪资信息等。

参数：响应时间应控制在 X 秒内，以确保数据的实时性；吞吐时间应满足每日至少处理 Y 条数据记录的需求。

异常条件：在数据输入错误或格式不符合规范时，CSCI 应能够自动提示并拒绝处理，确保数据的准确性。

目标：实现快速、准确的数据处理，确保人事资源信息的实时更新和高效利用。

方法和技术：采用高效的数据结构和算法，利用数据库管理系统进行数据的存储和检索。通过优化查询语句和索引设计，提高数据处理速度。同时，利用数据校验和验证机制，确保数据的准确性。

由来和背景：随着企业规模的扩大和人事资源信息的不断增加，传统的手工处理方式已无法满足需求。因此，需要借助计算机系统进行数据处理，以提高工作效率和准确性。

输入源：主要来源于企业内部的人事部门、考勤设备、薪资系统等，以及员工自行提交的数据（如请假申请、个人信息更新等）。

数量：数据量根据企业规模和员工数量而定，可能涉及数百万条甚至更多的记录。

度量单位：对于考勤数据，可能涉及的度量单位包括小时、分钟等；对于薪资数据，可能涉及的度量单位包括元、角、分等。

时间设定：系统支持实时数据输入，同时也可处理历史数据。对于实时数据，要求系统能够即时响应并处理；对于历史数据，可根据需求进行批量导入和处理。

有效输入范围：根据数据类型和字段定义，系统会设定相应的有效输入范围。例如，员工年龄应在合理范围内，薪资数据应符合企业的薪资标准等。对于超出有效范围的输入，系统将进行提示并拒绝处理。

输入数据的有效性检查：

验证输入数据的格式是否符合预定义的标准，例如日期格式、数字格式等。

检查输入数据是否在规定的有效范围内，如年龄、薪资等字段是否有合理的上下限。

验证输入数据是否完整，没有缺失必要的字段或信息。

操作的顺序：

首先接收输入数据。

接着进行数据有效性检查。

如果数据有效，则执行相应的数据处理操作（如查询、更新、删除等）。

最后输出处理结果。

异常情况的响应：

如果输入数据无效，则提示用户输入错误，并拒绝进一步处理。

在数据处理过程中，如果发生溢出、计算错误等异常情况，则记录错误日志，并通知用户或管理员进行处理。

如果遇到通信故障，尝试重新连接或通知用户稍后重试。

受操作影响的参数：

数据库中的相关记录会被更新或修改。

系统内部的状态变量或计数器可能会因操作而改变。

转换方法：

使用数据库查询语句对输入数据进行检索、更新或删除操作。

应用业务逻辑对数据进行计算、转换或格式化。

使用数据验证算法确保数据的准确性和一致性。

输出数据的有效性检查：

验证输出数据是否符合预期的格式和结构。

检查输出数据是否完整，没有遗漏重要信息。

如果输出数据涉及计算或汇总，验证其准确性和合理性。

输出目的地：输出数据通常存储于数据库或缓存中，以供其他模块或用户查询和使用。对于需要实时反馈的操作，输出数据也可能直接返回给调用方或展示在用户界面上。

数量：输出数据的数量取决于处理操作的类型和范围。例如，查询操作可能返回多条记录，而统计操作可能返回一个汇总值。

度量单位：输出数据的度量单位与输入数据相对应，确保数据的一致性和可读性。例如，薪资数据可能以元为单位，时间数据可能以小时或分钟为单位。

时间关系：输出数据的时间关系取决于处理操作的性质。对于实时处理，输出数据应反映当前状态或最新变化；对于历史数据处理，输出数据可能包含时间戳或时间段信息，以表示数据的有效期或发生时间。

有效输出范围：输出数据应符合业务逻辑和数据规范的有效范围。超出有效范围的数据应被视为异常或错误，并进行相应处理。

非法值的处理：如果输出数据包含非法值（如空值、无效字符等），系统应根据业务规则进行过滤、转换或提示错误。对于无法修正的非法值，应记录错误日志并通知相关人员进行处理。

出错信息：在数据处理过程中，如果出现错误或异常情况，系统应生成详细的出错信息，包括错误代码、错误描述和发生时间等。这些信息有助于定位问题原因并进行修复。

3.4.2 用户界面友好性：

需求：CSCI 应提供直观、易用的用户界面，方便用户进行各项操作。

参数：用户界面的响应时间应低于 X 秒，以减少用户等待时间；界面设计应符合通用设计原则，确保易于理解和操作。

异常条件：在用户操作失误时，CSCI 应提供明确的错误提示和建议，帮助用户快速纠正。

目标：提供直观、易用的用户界面，降低用户学习成本，提高用户满意度。

方法和技术：采用图形化界面设计，提供清晰的菜单和按钮，简化操作流程。同时，利用用户反馈和可用性测试，不断优化界面设计和交互方式。

由来和背景：良好的用户界面能够提升用户体验，减少操作错误，从而提高工作效率。此外，随着技术的不断发展，用户对软件界面的期望也在不断提高，因此需要不断优化界面设计以满足用户需求。

输入源：主要来源于用户的操作输入，包括鼠标点击、键盘输入等。

数量：输入数据的数量取决于用户的操作频率和复杂度，无法进行具体量化。

度量单位：对于用户界面而言，不涉及具体的度量单位。

时间设定：用户界面应实时响应用户的操作输入，确保操作的即时性和流畅性。

有效输入范围：用户界面会根据不同的操作和功能设定相应的有效输入范围。例如，在填写员工信息时，会限制输入字段的长度和格式；在进行搜索操作时，会设定搜索关键词的最小长度等。对于超出有效范围的输入，系统会进行提示并要求用户重新输入。

输入数据的有效性检查：

在用户输入时，实时检查输入内容是否符合要求，如格式、长度等。

对于需要选择或填写的字段，验证用户的选择或填写是否符合预设的选项或范围。

操作的顺序：

用户进行界面操作（如点击按钮、选择菜单项等）。

系统接收操作并检查输入数据的有效性。

如果输入有效，执行相应的界面更新或数据处理操作。

异常情况的响应：

如果用户输入无效，显示错误提示信息，并阻止进一步操作。

如果界面操作导致系统异常（如响应超时、界面卡顿等），则尝试恢复界面状态或提示用户重新尝试。

受操作影响的参数：

界面元素的显示状态或内容可能会因用户操作而改变。

与界面操作相关的系统变量或状态可能会更新。

转换方法：

使用图形界面库或框架来呈现和更新界面元素。

应用事件处理机制来响应用户的操作和输入。

输出数据的有效性检查：

对于界面上的显示内容，验证其是否符合预期的格式和布局。

如果界面操作涉及数据展示，验证展示的数据是否准确、完整。

输出目的地：用户界面的输出数据主要展示在屏幕上，供用户查看和操作。部分数据也可能通过声音、动画等形式进行输出。

数量：输出数据的数量取决于用户界面上展示的元素和信息的复杂度。例如，一个列表可能展示多条记录，一个图表可能包含多个数据点。

度量单位：对于需要展示度量单位的数据（如薪资、长度等），用户界面应清晰地显示相应的单位信息，以确保用户正确理解数据含义。

时间关系：对于涉及时间的数据（如考勤记录、任务截止时间等），用户界面应展示数据的时间戳或时间范围，以帮助用户了解数据的时效性和顺序。

有效输出范围：用户界面应根据业务需求和设计规范，限制输出数据的显示范围和格式。超出有效范围的数据应进行适当的处理或隐藏。

非法值的处理：对于从后端接收到的非法值或无效数据，用户界面应进行相应的错误处理，如显示错误提示、禁用相关操作按钮等。

出错信息：如果用户在进行界面操作时遇到错误或异常情况（如网络故障、输入验证失败等），用户界面应显示友好的出错信息，引导用户进行正确的操作或寻求帮助。

3.4.3 网络通信能力：

需求：CSCI 应具备稳定、可靠的网络通信能力，以实现数据的远程访问和共享。

参数：网络通信的响应时间应满足实时性的要求；数据传输的准确率和完整性应达到 99.9% 以上。

异常条件：在网络故障或中断时，CSCI 应具备自动重连或数据缓存功能，确保数据的完整性和安全性。

目标：实现数据的远程访问和共享，支持多用户并发操作，确保数据的一致性。

方法和技术：采用分布式系统架构，利用网络通信协议（如 HTTP、WebSocket 等）实现数据的传输和同步。同时，采用负载均衡和容错技术，确保系统的稳定性和可用性。

由来和背景：随着企业信息化程度的提高，越来越多的员工需要在不同地点和时间访问人事资源管理系统。因此，需要实现网络通信能力，以满足远程办公和移动办公的需求。

输入源：主要来源于网络请求，包括用户的登录请求、数据查询请求、数据更新请求等。

数量：网络请求的数量取决于用户的并发数量和操作频率，无法进行具体量化。

度量单位：对于网络通信而言，不涉及具体的度量单位。

时间设定：网络通信应实时响应用户的请求，确保请求的及时性和响应速度。

有效输入范围：网络通信会根据不同的请求类型和功能设定相应的有效请求格式和参数范围。对于不符合要求的请求，系统将进行提示并拒绝处理。

输入数据的有效性检查：

验证网络请求的格式是否符合协议标准，如 HTTP 请求头、请求体等。

检查请求参数是否有效和完整，如 API 密钥、数据字段等。

操作的顺序：

接收网络请求。

对请求进行有效性检查。

如果请求有效，执行相应的网络操作（如发送数据、接收响应等）。

处理响应并返回结果给客户端。

异常情况的响应：

如果请求无效，返回错误响应码和提示信息给客户端。

在网络通信过程中，如果遇到网络故障、超时或认证失败等情况，记录错误日志，并尝试重连或通知客户端。

受操作影响的参数：

网络连接的状态和参数可能会因通信操作而改变。

与网络通信相关的系统变量或计数器可能会更新。

转换方法：

使用网络通信库或框架来发送和接收网络请求。

应用协议解析和序列化机制来处理请求和响应数据。

输出数据的有效性检查：

验证网络响应的格式是否符合预期，如状态码、响应体等。

检查响应数据是否完整，没有丢失重要信息。

如果响应数据需要进行进一步处理或展示，验证其准确性和合理性。

输出目的地：网络通信的输出数据通常发送至远程客户端或服务器，以实现数据的共享和同步。输出数据可能通过网络协议进行封装和传输。

数量：输出数据的数量取决于通信请求的内容和目的。例如，一个查询请求可能返回多条记录作为输出数据。

度量单位：网络通信的输出数据一般不涉及特定的度量单位，除非数据本身具有明确的度量属性（如文件大小、传输速度等）。

时间关系：网络通信的输出数据应实时或按指定时间间隔发送，以确保数据的及时性和时效性。对于需要实时响应的应用场景，输出数据的传输延迟应尽可能短。

有效输出范围：网络通信的输出数据应符合协议标准和数据规范的有效范围。超出有效范围的数据可能导致通信失败或数据解析错误。

非法值的处理：在网络通信过程中，如果接收到非法值或无效数据，系统应进行错误处理，如丢弃无效数据、发送错误响应或请求重传等。

3.5 CSCI 外部接口需求

本条应分条描述 CSCI 外部接口的需求。(如有)本条可引用一个或多个接口需求规格说明(IRS)或包含这些需求的其他文档。

外部接口需求，应分别说明：

- a. 用户接口；
- b. 硬件接口；
- c. 软件接口；
- d. 通信接口的需求。

a. 用户接口

用户接口设计：

Web 界面：系统应提供一个直观、易于操作的 Web 界面，供用户进行登录、员工信息管理、薪资管理、考勤记录等操作。

响应式设计：界面应支持不同尺寸和分辨率的显示设备，确保在各种设备上都有良好的用户体验。

多语言支持：根据需求，系统应提供多语言切换功能，以满足不同用户的语言偏好。

交互与反馈：

操作提示：在用户进行关键操作时，系统应提供明确的操作提示，引导用户正确完成操作。

实时反馈：对于用户的操作，系统应实时给予反馈，如显示操作成功或失败的提示信息。

权限管理：

角色与权限：系统应支持角色和权限管理，不同角色的用户拥有不同的操作权限，确保数据的安全性和完整性。

登录认证：用户登录时应进行身份验证，确保只有合法用户才能访问系统。

b. 硬件接口

服务器硬件：

兼容性：系统应兼容常见的服务器硬件设备，如 CPU、内存、存储等。

性能要求：根据系统规模 and 用户数量，对服务器的性能进行合理配置，确保系统的稳定运行和高效响应。

网络硬件：

网络连接：系统应支持通过标准的网络设备进行连接，如路由器、交换机等。

网络带宽：根据系统的数据传输需求，合理配置网络带宽，确保数据的快速传输和实时响应。

c. 软件接口

前端与后端接口：

RESTful API：前端通过 RESTful API 与后端进行通信，实现数据的交互和操作。

数据格式：接口数据采用 JSON 格式进行传输，确保数据的可读性和解析效率。

数据库接口：

ORM 框架：后端使用 ORM 框架（如 Spring Data JPA）与 MySQL 数据库进行交互，实现数据的增删改查操作。

事务管理：对于涉及多个数据表的操作，系统应支持事务管理，确保数据的一致性和完整性。

第三方服务接口：

邮件服务：系统应集成邮件服务接口，用于发送通知、验证码等邮件。

短信服务：根据需求，系统可集成短信服务接口，用于发送短信通知或验证码。

d. 通信接口的需求

HTTP 通信:

协议: 系统采用 HTTP 或 HTTPS 协议进行通信, 确保数据的安全传输。

端口: 使用标准的 HTTP 端口 (如 80 或 443) 进行通信, 确保与其他系统的兼容性。

负载均衡与反向代理:

Nginx 配置: 系统可使用 Nginx 作为反向代理服务器, 实现后端服务器的负载均衡, 提高系统的可用性和性能。

SSL/TLS 加密: 对于敏感数据的传输, 系统应配置 SSL/TLS 加密, 保护数据在传输过程中的安全性。

接口安全性:

接口认证: 对于敏感或私有接口, 系统应实现接口认证机制 (如 OAuth、API 密钥等), 确保只有授权用户才能访问。

防止攻击: 系统应采取措施防止常见的网络攻击, 如 SQL 注入、跨站脚本攻击 (XSS) 等。

3.5.1 接口标识和接口图

本条应标识所需的 CSCI 外部接口, 也就是 CSCI 和与它共享数据、向它提供数据或与之交换数据的实体的关系。(若适用)每个接口标识应包括项目唯一标识符, 并应用名称、序号、版本和引用文件指明接口的实体(系统、配置项、用户等)。该标识应说明哪些实体具有固定的接口特性(因而要对这些接口实体强加接口需求), 哪些实体正被开发或修改(从而接口需求已施加给它们)。可用一个或多个接口图来描述这些接口。

3.5.x(接口的项目唯一标识符)

本条(从 3.5.2 开始)应通过项目唯一标识符标识 CSCI 的外部接口, 简单地标识接口实体, 根据需要可分条描述为实现该接口而强加于 CSCI 的需求。该接口所涉及的其他实体的接口特性应以假设或“当[未提到实体]这样做时, CSCI 将……”的形式描述, 而不描述为其他实体的需求。本条可引用其他文档(如: 数据字典、通信协议标准、用户接口标准)代替在此所描述的信息。(若适用)需求应包括下列内容, 它们以任何适合于需求的顺序提供, 并从接口实体的角度说明这些特性的区别(如对数据元素的大小、频率或其他特性的不同期望):

a.CSCI 必须分配给接口的优先级别;

b.要实现的接口的类型的需求(如: 实时数据传送、数据的存储和检索等);

c.CSCI 必须提供、存储、发送、访问、接收的单个数据元素的特性, 如:

1)名称/标识符;

a)项目唯一标识符;

b)非技术(自然语言)名称;

c)标准数据元素名称;

d)技术名称(如代码或数据库中的变量或字段名称);

e)缩写名或同义名;

2)数据类型(字母数字、整数等);

3)大小和格式(如: 字符串的长度和标点符号);

4)计量单位(如: 米、元、纳秒);

5)范围或可能值的枚举(如: 0-99);

6)准确度(正确程度)和精度(有效数字位数);

7)优先级别、时序、频率、容量、序列和其他的约束条件, 如: 数据元素是否可被更新和业

务规则是否适用；

8)保密性和私密性的约束；

9)来源(设置/发送实体)和接收者(使用/接收实体)；

d.CSCI 必须提供、存储、发送、访问、接收的数据元素集合体(记录、消息、文件、显示和报表等)的特性，如：

1)名称/标识符；

a)项目唯一标识符；

b)非技术(自然语言)名称；

c)技术名称(如代码或数据库的记录或数据结构)；

d)缩写名或同义名；

2)数据元素集合体中的数据元素及其结构(编号、次序、分组)；

3)媒体(如盘)和媒体中数据元素/数据元素集合体的结构；

4)显示和其他输出的视听特性(如：颜色、布局、字体、图标和其他显示元素、蜂鸣器以及亮度等)；

5)数据元素集合体之间的关系。如排序/访问特性；

6)优先级别、时序、频率、容量、序列和其他的约束条件，如：数据元素集合体是否可被修改和业务规则是否适用；

7)保密性和私密性约束；

8)来源(设置/发送实体)和接收者(使用/接收实体)；

e.CSCI 必须为接口使用通信方法的特性。如：

1)项目唯一标识符；

2)通信链接/带宽/频率/媒体及其特性；

3)消息格式化；

4)流控制(如：序列编号和缓冲区分配)；

5)数据传送速率，周期性/非周期性，传输间隔；

6)路由、寻址、命名约定；

7)传输服务，包括优先级别和等级；

8)安全性/保密性/私密性方面的考虑，如：加密、用户鉴别、隔离和审核等；

f.CSCI 必须为接口使用协议的特性，如：

1)项目唯一标识符；

2)协议的优先级别/层次；

3)分组，包括分段和重组、路由和寻址；

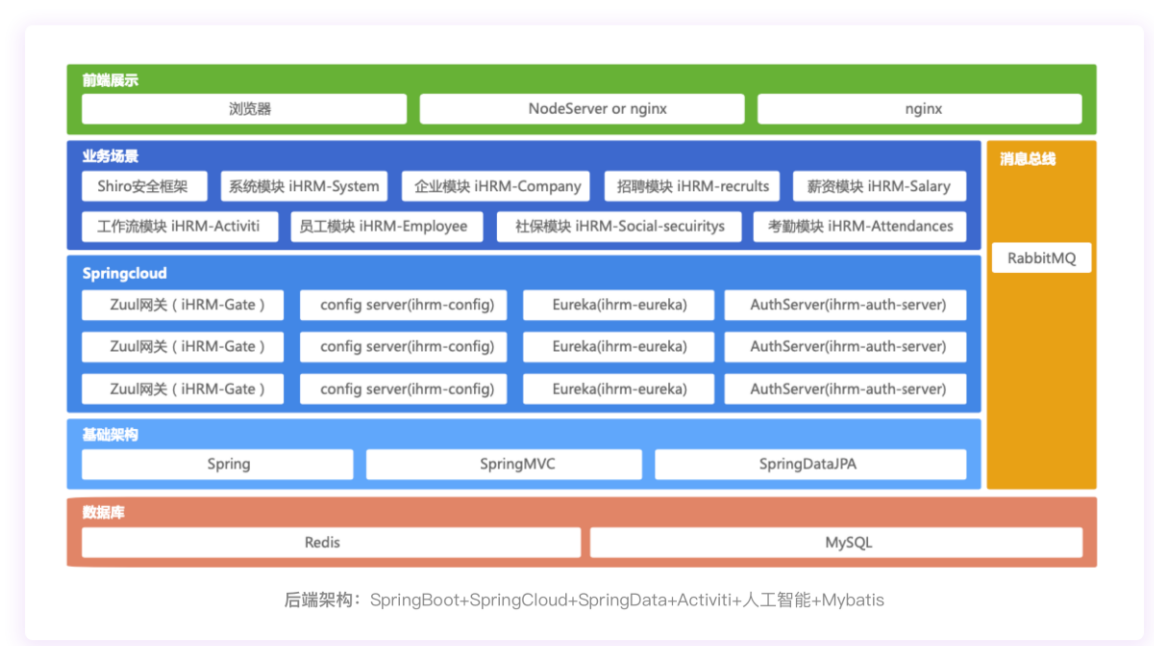
4)合法性检查、错误控制和恢复过程；

5)同步，包括连接的建立、维护和终止；

6)状态、标识、任何其他的报告特征；

g.其他所需的特性，如：接口实体的物理兼容性(尺寸、容限、负荷、电压和接插件兼容性等)。

整体结构：



接口描述：

图形接口采用了图形化的操作界面，用非常容易识别的各种图标来将系统各项功能、各种应用程序和文件，直观、逼真地表示出来。用户可通过鼠标、菜单和对话框来完成对应用程序和文件的操作。图形用户接口元素包括窗口、图标、菜单和对话框，图形用户接口元素的基本操作包括菜单操作、窗口操作和对话框操作等。

当涉及到与后端进行数据交互时，通常会使用 **Vue 3** 搭配其他库或技术来实现图形接口的调用和数据传输。一般来说，这包括以下内容：

图形接口实现：通常情况下，图形接口（**Graphical Interface**）实际上指的是 **API**（**Application Programming Interface**），也就是应用程序的编程接口。在 **Vue 3** 中，你可以使用 **JavaScript** 的 **Fetch API**、**Axios** 或者其他网络请求库来实现与后端 **API** 的交互。

图形接口入口：图形接口的入口通常是后端应用程序的端点（**Endpoint**），它们可以是 **RESTful API**、**GraphQL** 等。在 **Vue 3** 中，你可以通过调用这些端点来获取数据或发送请求。

数据传输格式：数据传输格式通常是通过 **HTTP** 请求和响应进行传输的，常见的格式包括 **JSON**、**XML** 等。在 **Vue 3** 中，你可以使用 **JavaScript** 的内置方法或者第三方库来处理这些数据格式。

3.6CSCI 内部接口需求

所有内部接口都留待设计时决定。

3.7CSCI 内部数据需求

一、员工基本信息

员工编号：唯一标识每个员工的编号，用于系统内部的查询和引用。

姓名：员工的全名。

性别：记录员工的性别信息。

出生日期：记录员工的出生年月日。

身份证号：记录员工的身份证号码，用于身份验证和合法性检查。

联系方式：包括电话号码、电子邮箱等，便于与员工进行沟通和联系。

二、教育背景与资质

学历：记录员工的最高学历及毕业院校。

专业：记录员工的专业背景。

培训经历：记录员工参加过的培训课程及取得的相关证书。

职业资格：记录员工所持有的职业资格证书或技能等级。

三、工作履历与职务

入职日期：记录员工的入职时间。

部门：记录员工所属的部门或团队。

职位：记录员工的职位或职务。

工作职责：描述员工的主要工作职责和任务。

晋升记录：记录员工的晋升历史及时间。

四、薪资与福利

基本工资：记录员工的月薪或年薪。

奖金与津贴：记录员工获得的各类奖金、津贴及发放标准。

社保与公积金：记录员工的社保缴纳情况及公积金缴纳比例。

福利待遇：描述员工享受的福利待遇，如年假、带薪休假等。

五、考勤与绩效

考勤记录：记录员工的上下班时间、请假、迟到、早退等情况。

绩效评估：记录员工的绩效评估结果及评价意见。

奖惩记录：记录员工获得的奖励或受到的处罚情况。

六、其他信息

家庭情况：记录员工的家庭成员信息，如配偶、子女等。

紧急联系人：记录员工的紧急联系人及联系方式。

备注信息：用于记录员工的特殊情况或需要关注的事项。

七、数据库与数据文件需求

数据库设计：系统应采用关系型数据库，确保数据的完整性、一致性和安全性。数据库应合理设计表结构、字段类型和索引，以提高数据查询和处理效率。

数据备份与恢复：系统应定期备份数据库，确保数据的可靠性和可恢复性。同时，应提供数据恢复机制，以便在数据丢失或损坏时能够迅速恢复。

数据文件存储：对于需要存储的文件（如员工照片、合同扫描件等），系统应提供文件上传、下载和管理的功能。文件应存储在安全可靠的位置，并设置适当的访问权限。

3.8 适应性需求

一、安装数据相关需求

地域适应性：系统应支持多地区、多语言设置，以适应不同地域的文化和语言习惯。

组织架构适应性：系统应能灵活配置不同的组织架构，包括部门、岗位、层级等，以适应不

同企业的组织结构。

数据格式兼容性：系统应支持多种数据格式的导入和导出，如 Excel、CSV 等，以方便用户迁移数据和与其他系统进行数据交互。

二、运行参数相关需求

用户权限管理：系统应提供灵活的用户权限配置功能，根据用户的角色和职责设置不同的权限级别，以确保数据的安全性和系统的稳定运行。

流程定制化：系统应支持流程的自定义配置，如招聘流程、入职流程、离职流程等，以适应企业不同的业务需求。

报表生成与自定义：系统应提供丰富的报表生成功能，并支持用户自定义报表模板和字段，以满足不同层级的决策分析需求。

三、其他适应性需求

系统扩展性：系统应具备良好的扩展性，能够随着企业业务的发展而不断升级和扩展功能。

界面自定义：系统应支持界面自定义功能，用户可以根据个人喜好和工作习惯调整界面布局和样式。

性能优化：系统应能够根据用户的使用情况和数据量进行性能优化，确保在高并发、大数据量的情况下仍能保持稳定运行。

3.9 保密性需求

数据访问控制：

系统应实施严格的身份验证和权限管理机制，确保只有经过授权的用户才能访问相关的人事信息。

根据用户的角色和职责，分配不同的数据访问权限，如员工基本信息查看、薪资数据修改等。实行最小权限原则，即用户只能访问其完成工作所必需的数据，不能越权访问其他敏感信息。

数据加密：

对存储在系统中的敏感数据（如身份证号、银行账号等）进行加密处理，以防止数据泄露。在数据传输过程中，使用安全的通信协议（如 HTTPS）进行加密传输，确保数据在传输过程中的安全。

审计和日志记录：

系统应记录所有对敏感数据的访问和修改操作，包括访问时间、操作人、操作内容等，以便于后续的审计和追踪。

定期对日志进行审查和分析，及时发现潜在的安全威胁和违规行为。

防止数据泄露：

系统应采取防止数据泄露的措施，如禁止直接下载或导出敏感数据为明文格式。

对于需要共享或传递的数据，应通过安全的方式进行处理，如使用安全的文件共享平台或加密的邮件传输。

系统安全加固：

定期对系统进行安全漏洞扫描和风险评估，及时修复已知的安全漏洞。

采用防火墙、入侵检测系统等安全措施，增强系统的整体安全防护能力。

应急响应和灾难恢复：

制定完善的应急响应计划，包括数据安全事件的处理流程、联系人和联系方式等，确保在发生安全事件时能够迅速响应和处理。

建立数据备份和恢复机制，确保在发生数据丢失或损坏时能够迅速恢复数据。

3.10 保密性和私密性需求

3.10.1 保密性需求

1. CSCI 运行的保密性环境

系统应部署在安全的网络环境中，采用物理隔离或逻辑隔离措施，确保与外部网络的隔离。访问系统的终端设备应受到访问控制和安全策略管理，防止未经授权的访问。

2. 提供的保密性类型和程度

系统应实现数据的保密性，确保敏感数据（如员工个人信息、薪资数据等）在存储、处理和传输过程中不被未经授权的用户获取。

系统应支持不同级别的保密性要求，如针对不同角色或敏感程度的数据设置不同的访问权限。

3. CSCI 必须经受的保密性风险

系统应能够抵御来自外部和内部的保密性威胁，如黑客攻击、恶意软件、内部人员泄露等。系统应定期评估潜在的安全风险，并采取相应的安全措施进行防范。

4. 减少保密性风险的安全措施

采用强密码策略，定期更换密码，并限制密码尝试次数。

使用加密技术保护敏感数据的存储和传输。

定期进行安全漏洞扫描和风险评估，及时修复已知的安全漏洞。

5. CSCI 必须遵循的保密性政策

系统应遵循国家和行业相关的保密性法律法规，如《个人信息保护法》等。

制定并执行系统内部的保密性政策，明确员工的信息保密义务和责任。

6. 保密性审核

系统应定期进行保密性审核，检查系统的保密性措施是否得到有效执行。

审核结果应作为改进系统保密性的依据，及时调整和优化保密措施。

3.10.2 私密性需求

1. CSCI 运行的私密性环境

系统应提供用户友好的隐私设置选项，允许员工控制其个人信息的可见性和共享范围。

确保员工在使用系统时，其个人信息和操作记录不被其他用户或系统管理员无端窥探。

2. 提供的私密性类型和程度

系统应确保员工个人信息的私密性，包括个人身份信息、联系方式、家庭状况等。

根据员工的需求和偏好，提供个性化的隐私设置选项，以满足不同用户的私密性要求。

3. CSCI 必须遵循的私密性政策

系统应遵循国家和行业相关的隐私保护法律法规，如《隐私政策》等。

制定并执行系统内部的隐私政策，明确收集、使用、存储和共享员工个人信息的规则和限制。

4. 私密性审核

系统应定期进行私密性审核，检查系统是否遵循了相关的隐私政策和法规。

审核结果应作为改进系统私密性的依据，确保员工的个人信息得到妥善保护。

5. 私密性确证/认可准则

系统应通过相关的隐私保护和信息安全认证，如 ISO 27001、ISO 27701 等，以证明其符合私密性和保密性的国际标准。

与第三方机构合作进行隐私保护评估，确保系统的私密性措施得到有效实施和认可。

3.11 CSCI 环境需求

本系统基于 SaaS 的 IHRM 系统，其环境需求主要包括以下几个方面：

前端环境：采用 Vue.js 框架进行开发，确保在主流浏览器上的兼容性，并利用 Webpack 或 Vite 进行项目构建与打包。

后端环境：使用 Spring 全家桶（包括 Spring Boot、Spring Security、Spring Data JPA 等）进行后端开发，提供 RESTful API 接口，保障系统的安全性和性能。

数据库环境：选用 MySQL 作为数据库管理系统，设计合理的数据库表结构，并利用索引优化查询性能。同时，采用 Navicat 作为可视化工具进行数据库管理和操作。

负载均衡：使用 nginx 作为反向代理服务器和负载均衡器，提高系统的可扩展性和稳定性。

3.12 计算机资源需求

为确保系统的正常运行和性能需求，需要以下计算机资源：

硬件资源：高性能的服务器配置，包括足够的 CPU、内存和存储空间，以支撑系统的运行和数据处理需求。

软件资源：安装并配置稳定的操作系统（如 Linux）、Java 环境和 MySQL 数据库管理系统，以提供稳定的运行环境。

3.12.1 计算机硬件需求

为了满足人事管理系统的运行需求，我们需要以下硬件设备：

服务器：选择高性能多核处理器、具备高可靠性的机架式服务器，以支撑系统的核心运算和数据存储。服务器处理器至少应采用 8 核以上的最新一代处理器，并支持虚拟化技术，以便更高效地利用硬件资源。

工作站：为开发和测试团队配置工作站，这些工作站应至少配备中高级别的处理器、图形卡和大容量内存，以确保开发和测试工作的顺利进行。

存储设备：为保障数据的完整性和安全性，我们应选择配置有 RAID 技术的企业级存储阵列。此外，还需要配备标准的 USB 接口键盘和鼠标、至少 24 英寸高分辨率显示器，以及 USB 3.0 接口的外部硬盘用于数据备份。

3.12.2 计算机硬件资源利用需求

为了确保系统的稳定运行和高效性能，我们需要对硬件资源的利用进行以下规划：

处理器能力：服务器的处理器最大许可使用能力不应超过 80%，工作站则不超过 70%，以保留足够的处理能力应对突发负载。

存储器容量：服务器内存使用不应超过 85%，工作站内存使用不超过 75%，以确保系统有足够的内存资源进行数据处理。

输入/输出设备能力：系统应保持正常的 I/O 吞吐量，避免出现明显的延迟现象。

辅助存储器容量：外部硬盘应保持至少 20% 的可用空间，以便进行数据备份和其他临时存

储需求。

3.12.3 计算机软件需求

软件方面，我们需要以下配置来支持系统的运行和开发工作：

操作系统：服务器将采用 Linux 发行版以确保稳定性和安全性，而工作站则使用 Windows 10 专业版以满足开发和测试的需求。

数据库管理系统：我们将采用 Oracle 或 MySQL 等成熟的数据库管理系统来存储和管理数据。

通信/网络软件：系统需要配置标准的 TCP/IP 协议栈，并支持 SSH 和 VPN 远程访问，以便进行安全的远程管理和维护。

实用软件：为了满足开发需求，我们需要配置集成开发环境（IDE）和版本控制软件（如 Git）。

3.12.4 计算机通信需求

为了确保系统的通信效率和稳定性，我们需要考虑以下方面：

地理位置：系统需要连接多个地理位置的服务器和工作站，形成一个分布式的网络环境。

配置和网络拓扑结构：我们将采用星型拓扑结构来组织网络，并配置防火墙和入侵检测系统以增强安全性。

传输技术：光纤通道或高速以太网传输技术将被用于确保数据的高速传输。

数据传输速率：系统应保持稳定的 1Gbps 或以上的数据传输速率，以满足大规模数据处理和传输的需求。

3.13 软件质量因素

可靠性：确保系统长时间稳定运行，减少故障发生的概率。

可维护性：设计清晰的代码结构和完善的文档，方便后续的系统维护和功能扩展。

性能：优化系统性能，确保快速响应请求和处理大量数据。

安全性：采取严格的安全措施，保护用户数据和系统的安全性。

3.14 设计和实现的约束

在设计和实现过程中，需要遵循以下约束条件：

MVC 设计模式：采用 MVC 设计模式实现前后端分离，降低系统的耦合度。

RESTful API：使用 RESTful API 进行前后端通信，确保接口的一致性和可扩展性。

编码规范：遵循 Java 编码规范和最佳实践，提高代码的可读性和可维护性。

数据库设计：遵循 MySQL 数据库设计和优化原则，确保数据的完整性和查询性能。

3.15 数据

本系统需要设计合理的数据模型，确保数据的完整性和一致性。同时，采用 ORM 框架进行

数据库操作，提高开发效率。对于敏感数据，需要进行加密存储和传输，保障数据的安全性。

3.16 操作

系统需要提供用户友好的操作界面和流程，方便用户进行各项操作。同时，设计合理的权限管理系统，控制用户的操作权限。此外，还需要提供操作日志记录功能，方便问题的追踪和审计。

3.17 故障处理

故障处理是确保系统稳定运行的关键环节。我们需要识别并分类可能发生的软硬件故障，如内存泄漏、网络中断等，并区分故障的严重程度。同时，建立详细的故障日志记录机制，以便快速定位和解决问题。恢复策略应包括自动和手动两种方式，并对于可预见的故障设计预防措施和监控系统。

3.18 算法说明

在系统开发中，选择合适的算法至关重要。我们会根据需求挑选高效的排序、搜索或加密算法，并详细描述每个主要算法的工作原理和关键步骤。此外，我们会通过充分的测试和验证来确保算法的正确性和可靠性，并根据实际运行情况进行优化。

1.员工信息检索算法：概况：该算法用于从数据库中检索员工的基本信息，如姓名、职位、部门、入职日期等。它可以根据不同的查询条件（如姓名、部门、职位等）进行筛选和排序。详细公式：检索结果 = SELECT 字段列表 FROM 员工表 WHERE 检索条件。

2.薪资计算算法：概况：此算法根据员工的薪资标准、出勤情况、绩效奖金等因素，计算出员工应得的薪资。它可能涉及到复杂的计算规则，如加班费、扣税、社保扣除等。详细公式：总薪资 = 基本薪资 + (加班小时数 * 加班费率) + 绩效奖金 - (社保费率 * 基本薪资) - 个税。

3.绩效考核算法：概况：该算法用于评估员工的工作绩效，通常基于多个维度（如工作质量、工作效率、团队合作等）进行打分和排名。绩效考核的结果可能影响到员工的晋升、薪资调整等方面。详细公式：总得分 = (工作质量得分 * 工作质量权重) + (工作效率得分 * 工作效率权重) + (团队合作得分 * 团队合作权重)。

4.排班调度算法：概况：这个算法用于制定员工的工作排班表，确保每个岗位都有合适的人员覆盖，同时考虑到员工的休假、轮班等需求。在复杂的组织结构中，排班调度可能需要考虑多种因素，如员工的工作经验、技能水平等。详细公式：目标函数：最小化排班不公平性、连续性中断等指标。约束条件：每个岗位必须有合格的人员覆盖；员工的工作时间、休息时间等必须符合国家法规和公司政策；考虑员工的个人偏好和技能水平。

3.19 有关人员需求

为了确保项目的顺利进行，我们会根据系统的规模和复杂度合理配置开发、测试和运维人员。我们会明确每个角色的职责和技能要求，并定期为他们提供技能培训，以提高团队的整体技术水平和创新能力。同时，建立良好的团队协作机制，促进团队成员间的有效沟通。

3.20 有关培训需求

为了让用户能够充分利用系统，我们会设计针对性的培训内容，包括系统操作、功能使用以及故障排除等。培训将采用线上和线下相结合的方式，以满足不同用户的需求。此外，我们还会提供详细的培训资料和视频教程，方便用户进行自主学习。为了确保培训效果，我们会建立评估机制，通过考试、问卷调查等方式收集用户反馈，并据此不断改进培训内容和方法。

3.21 有关后勤需求

为确保系统的正常运行，需要提供稳定可靠的电力和网络环境。同时，为开发团队提供必要的办公设备和软件工具，方便其进行开发工作。此外，建立完善的文档管理系统，方便团队成员查阅和共享相关文档。

3.22 其他需求

- 1.安全需求：这是确保 CSCI 在运行过程中能够抵御各种安全威胁的需求，如未经授权的访问、数据泄露或恶意攻击。例如，系统可能需要实现强密码策略、访问控制和数据加密等功能。
- 2.性能需求：这涉及到 CSCI 在特定工作负载下的性能表现，如响应时间、吞吐量等。例如，系统可能需要保证在高峰时段能够处理大量的用户请求，而不会出现明显的延迟或崩溃。
- 3.兼容性需求：如果 CSCI 需要与其他软件或系统协同工作，那么就需要满足特定的兼容性要求。例如，系统可能需要与其他企业的系统进行数据交换，这就要求系统能够支持特定的数据格式和通信协议。
- 4.法律合规需求：CSCI 的设计和实现必须符合相关的法律法规和标准。例如，如果系统处理的是个人数据，那么就需要遵守相关的数据保护法规。

3.23 包装需求

无。

3.24 需求的优先次序和关键程度

根据业务需求和技术实现难度，确定各项需求的优先次序和关键程度。对于关键需求进行重点关注和测试，确保系统的核心功能得到满足。同时，根据实际情况调整需求的优先级和关键程度，以适应业务变化和用户需求的变化。

4 合格性规定

本章定义一组合格性方法，对于第 3 章中每个需求，指定所使用的方法，以确保需求得到满足。可以用表格形式表示该信息，也可以在第 3 章的每个需求中注明要使用的方法。合格性方法包括：

- a.演示：运行依赖于可见的功能操作的 CSCI 或部分 CSCI,不需要使用仪器、专用测试设备或进行事后分析；
- b.测试：使用仪器或其他专用测试设备运行 CSCI 或部分 CSCI,以便采集数据供事后分析使用；
- c.分析：对从其他合格性方法中获得的积累数据进行处理，例如测试结果的归约、解释或推断；
- d.审查：对 CSCI 代码、文档等进行可视化检查；
- e.特殊的合格性方法。任何应用到 CSCI 的特殊合格性方法，如：专用工具、技术、过程、设施、验收限制。

需求类别	需求描述	合格性方法
CSCI 能力需求	确保 CSCI 满足系统功能和性能要求	a. 演示：展示 CSCI 的关键功能
		b. 测试：对 CSCI 进行功能测试和性能测试
		c. 分析：对测试结果进行归约和解释，验证是否满足要求
CSCI 外部接口需求	验证 CSCI 与外部系统或设备的交互能力	a. 演示：展示 CSCI 与外部系统的集成效果
		b. 测试：进行接口测试，确保数据交互的正确性和稳定性
		c. 分析：对接口数据进行事后分析，检查是否存在异常或错误
CSCI 内部接口需求	确保 CSCI 内部组件或模块之间的交互顺畅	d. 审查：检查内部接口的代码实现和文档描述
		e. 特殊的合格性方法：使用专用工具或技术进行内部接口的静态和动态分析
CSCI 内部数据需求	验证 CSCI 内部数据处理和存储的正确性	a. 演示：展示数据处理流程的正确性
		b. 测试：对数据进行单元测试或集成测试，确保数据的完整性和准确性
		c. 分析：对测试数据进行事后分析，检查数据

需求类别	需求描述	合格性方法
		处理的逻辑是否正确
适应性需求	确保 CSCI 能够适应不同的环境和条件	b. 测试：在不同环境和条件下对 CSCI 进行测试，验证其适应性
		e. 特殊的合格性方法：使用模拟环境或特殊设施进行适应性测试
保密性需求	保证 CSCI 的数据和信息安全，防止未经授权的访问	d. 审查：对 CSCI 的安全代码和配置进行审查
		e. 特殊的合格性方法：使用专业的安全测试工具和技术，对 CSCI 进行安全漏洞扫描和风险评估
私密性需求	确保 CSCI 在处理个人敏感信息时符合隐私保护要求	d. 审查：检查 CSCI 的隐私保护政策和实施情况
		e. 特殊的合格性方法：使用隐私保护评估工具，对 CSCI 进行隐私保护能力的评估
CSCI 环境需求	验证 CSCI 在特定环境下的运行能力和稳定性	b. 测试：在目标环境下对 CSCI 进行压力测试、稳定性测试等
		e. 特殊的合格性方法：使用专用的环境模拟工具或设施，对 CSCI 进行环境适应性测试
计算机资源需求	确保 CSCI 在运行过程中合理利用计算机资源	b. 测试：对 CSCI 进行性能测试和资源消耗测试
		c. 分析：对测试数据进行归约和解释，分析 CSCI 的资源利用效率

5 需求可追踪性

a. CSCI 需求到系统需求的可追踪性

本节描述从计算机软件配置项（CSCI）的需求到其所涉及的系统（或子系统）需求的可追踪性。这种追踪性可以通过在第三章中对每个需求进行注释的方式来描述。

例如，如果一个 CSCI 需求是基于系统体系结构设计中产生的接口需求，而这些接口需求在更高层次的系统需求文档中没有直接提及，那么这个 CSCI 需求可以追踪到如“系统实现”这样的一般需求，或追踪到导致它们产生的具体系统设计决策。

表 5.1: CSCI 需求到系统需求的追踪性

CSCI 需求编号	系统需求编号	追踪性描述
CSCI_REQ_01	SYS_REQ_03	CSCI 需求 01 支持系统需求 03 的实现
CSCI_REQ_02	SYS_REQ_07	CSCI 需求 02 追踪到系统实现决策

b. 系统需求到 CSCI 需求的可追踪性

本节描述从系统（或子系统）需求到被本规格说明中的 CSCI 的需求的可追踪性。所有分配到 CSCI 的系统（或子系统）需求都应在此加以说明，并提供追踪到的 CSCI 需求的引用。

例如，如果一个系统需求被分配给了特定的 CSCI，并且该需求在 CSCI 需求文档中有直接对应的条目，那么这个系统需求可以直接追踪到该 CSCI 需求。

表 5.2: 系统需求到 CSCI 需求的追踪性

系统需求编号	CSCI 需求编号	追踪性描述
SYS_REQ_03	CSCI_REQ_01	系统需求 03 由 CSCI 需求 01 实现
SYS_REQ_07	CSCI_REQ_02	系统需求 07 追踪到 CSCI 设计决策

注释：需求追踪性将通过需求追踪矩阵（RTM）进行管理，以确保需求之间的一致性和完整性。对于未在 IRS 中直接提及的 CSCI 需求，应在 RTM 中提供适当的引用和说明。

6 尚未解决的问题

1. 安全性问题：人事管理系统可能存在安全漏洞，如未经授权的访问、数据泄露、跨站脚本攻击（XSS）或 SQL 注入等问题。

性能问题：系统可能在处理大量数据或同时处理多个请求时出现性能问题，导致响应速度下降或系统崩溃。

. 用户体验问题：界面可能不够直观友好，导致用户操作困难或繁琐。

???. 功能完善性：系统可能缺乏一些必要的功能，如报表生成、自定义查询、权限管理等。

???. 兼容性问题：系统可能在不同浏览器或设备上表现不一致，或者对于特定操作系统存在兼容性问题。