

天津大学

软件设计文档



学院专业： 智算学部软件工程

年 级： 2018 级

组长姓名： 张豪

组长学号： 3018216216

组员姓名： 卫林泉

组员学号： 3018216147

组员姓名： 高喆妍

组员学号： 3018207022

组员姓名： 吴海鹰

组员学号： 3018216274

指导老师： 李罡

2020 年 10 月 27 日

变更记录

版本号	修改点说明	变更日期	变更人	审批人
0.0.1	创建、信息管理	2020/10/27	张豪	卫林泉
0.0.2	添加考勤管理	2020/10/30	卫林泉	张豪
1.0.0	请假管理	2020/11/3	高喆妍	卫林泉
1.0.1	接口、数据库设计	2020/11/3	吴海鹰	张豪

修改点说明的内容有如下几种：创建、修改（+修改说明）、删除（+删除说明）

目录

目录.....	3
1.前言.....	5
1.1 文档目的.....	5
1.2 文档范围.....	5
1.3 读者对象.....	5
1.4 参考文档.....	5
1.5 术语说明.....	5
2.总体设计.....	6
2.1 系统描述.....	6
2.1.1 系统规格概述.....	6
2.1.2 运行环境规定.....	6
2.1.3 数据库管理要求.....	7
2.1.4 故障处理要求.....	7
2.1.5 其他要求.....	7
2.2 设计原则.....	8
2.3 总体结构.....	9
3.详细设计.....	11
3.1 信息管理.....	11
3.1.1 功能概述.....	11
3.1.2 业务流程.....	12
3.1.3 创建账号.....	12
3.1.4 查询账号.....	13
3.1.5 修改信息.....	14
3.2 考勤管理.....	14
3.2.1 功能概述.....	14
3.2.2 业务流程.....	15
3.2.3 员工打卡.....	15
3.2.4 员工查看打卡信息.....	16
3.3 请假管理.....	17
3.3.1 功能概述.....	17
3.3.2 业务流程.....	17

3.3.3 提交请假.....	18
3.3.4 查看打卡.....	18
3.3.5 审核请假.....	19
3.3.6 查询剩余假期.....	20
4.接口设计.....	21
4.1 数据库设计规范.....	21
4.1.1 表设计规范.....	21
4.1.2 字段长度规则.....	21
4.2 URL 和方法命名规范.....	22
4.2.1 RESTFUL URL 命名规范.....	22
4.3 方法命名规范.....	23
4.3.1 DAO.....	23
4.3.2 Service.....	23
4.3.3 Rest.....	24
5.数据库设计.....	24
5.1 表结构.....	24
5.1.1 用户表（user）.....	24
5.1.2 考勤表（user_attendance）.....	25
5.1.3 请假表（user_leave）.....	25
5.1.4 部门表（department）.....	26
6.环境配置.....	26
6.1 开发环境配置.....	26
6.2 运行环境配置.....	26
6.3 测试环境配置.....	26

1.前言

1.1 文档目的

详细设计是根据考勤与请假系统《软件需求规格说明书》进一步分析、归纳，设计实现需求的技术策略。并进一步细化技术框架和内容，对下一阶段的工作提供指导。为明确软件体系结构、实现技术、部署方案，与客户对系统功能需求及实现方式达成一致的认识，以便于安排项目进度、组织软件开发与测试，特撰写本文档。

1.2 文档范围

本文档的读者为业主、用户、数据库建设人员、系统设计人员、开发人员和测试人员。

1.3 读者对象

- 设计人员：清楚与确认项目需求，并进行合理的系统设计。
- 开发人员：明确任务与需求，清楚项目开发流程。
- 测试人员：据此提取测试用例。

1.4 参考文档

- 《考勤与请假需求规格说明书》
- 《高级实践大作业 参考题目》

1.5 术语说明

表1. 术语说明

序号	名称	说明

2.总体设计

2.1 系统描述

2.1.1 系统规格概述

考勤与请假系统采用 B/S 架构，前后端分离的实现方案，前端使用 vue+elementUI 的技术架构,后端使用 SpringBoot 架构实现,数据库采用 MySQL。

2.1.2 运行环境规定

（1）硬件环境

服务器类型	数量	配置信息
数据库服务器（本地）	1	4 核，8GB 内存，512GB 存储
文档服务器（本地）	1	4 核，8GB 内存，512GB 存储
web 服务器（本地）	1	4 核，8GB 内存，512GB 存储
应用服务器（本地）	1	4 核，8GB 内存，512GB 存储

（2）软件环境

名称	版本及参数	用途
系统操作系统	Linux(CentOS7.4) open ssh 7.0 及其以上	系统部署环境
客户操作系统	Winxp,Win7,Win10	客户端系统
数据库软件	MySQL	提供系统的数据库服务
内存数据库	Redis	缓存服务

客户端浏览器	Chrome（版本 69 及以上）， 360 安全浏览器 10 以上， 360 极速浏览器	客户端系统访问
--------	---	---------

2.1.3 数据库管理要求

采用 MySQL 数据库，Redis 作为缓存。

2.1.4 故障处理要求

产品支撑环境运行错误，包括以下几个方面：

数据库运行错误：系统无法使用，将提供错误提示页面，并记录日志。

系统配置参数错误：系统提供错误提示页面，要求管理员修改相关配置参数。配置参数的修改方法应参考相关为管理员提供的手册。

系统参数初始化错误：提供系统初始化功能，用于成批完成系统初始化。在初始化过程中，对于严重影响平台运行的错误，提供错误提示页面。对于普通错误，记录日志。

应用程序错误：由应用程序提供错误提示页面，相关错误应记录日志。

2.1.5 其他要求

先进性：采用先进成熟的技术，确保系统的先进性、经济性和实用性。

安全可靠：提供的应用框架及平台本身提供应用安全保证，并可以和第三方安全手段，如认证、加密、电子签名等进行集成。必须保证数据的安全性和保密性。对于基于平台开发的应用系统，只允许有权限的人员进行操作和浏览信息。必须有安全的手段来进行权限控制。

开放互连：系统应对各类业务系统、数据库系统、WEB 信息等具有通用的或可定制的接口策略和连接方法。

规范性：开发过程控制、开发技术、系统编码、文档应规范化，并遵循相应的国内外标准。开发结束，需要提供必要的文档资料。

可靠性：保证系统的可靠运行和在升级过程中的方便快捷。

可扩充性：系统应当可以根据需求的变化，方便地进行功能的调整、增减，模块的升级和系统架构的逐步完善。

界面友好、操作方便：操作界面要直观、简单、贴近实际，操作过程应当尽量简化，符合实际过程。身份认证过程即要保证安全，也要尽量简化认证过程。

可维护性：系统维护应当简单。

2.2 设计原则

在对项目的规划设计过程中，根据项目建设目的及其相关功能，我们将遵循以下原则对项目进行规划设计：

（1） 统一规划，分步实施

考勤与请假系统，分为考勤，请假和信息管理三个模块，模块统筹应该注意高类聚低耦合等原则，应该统一规划好后再进行分布开发。

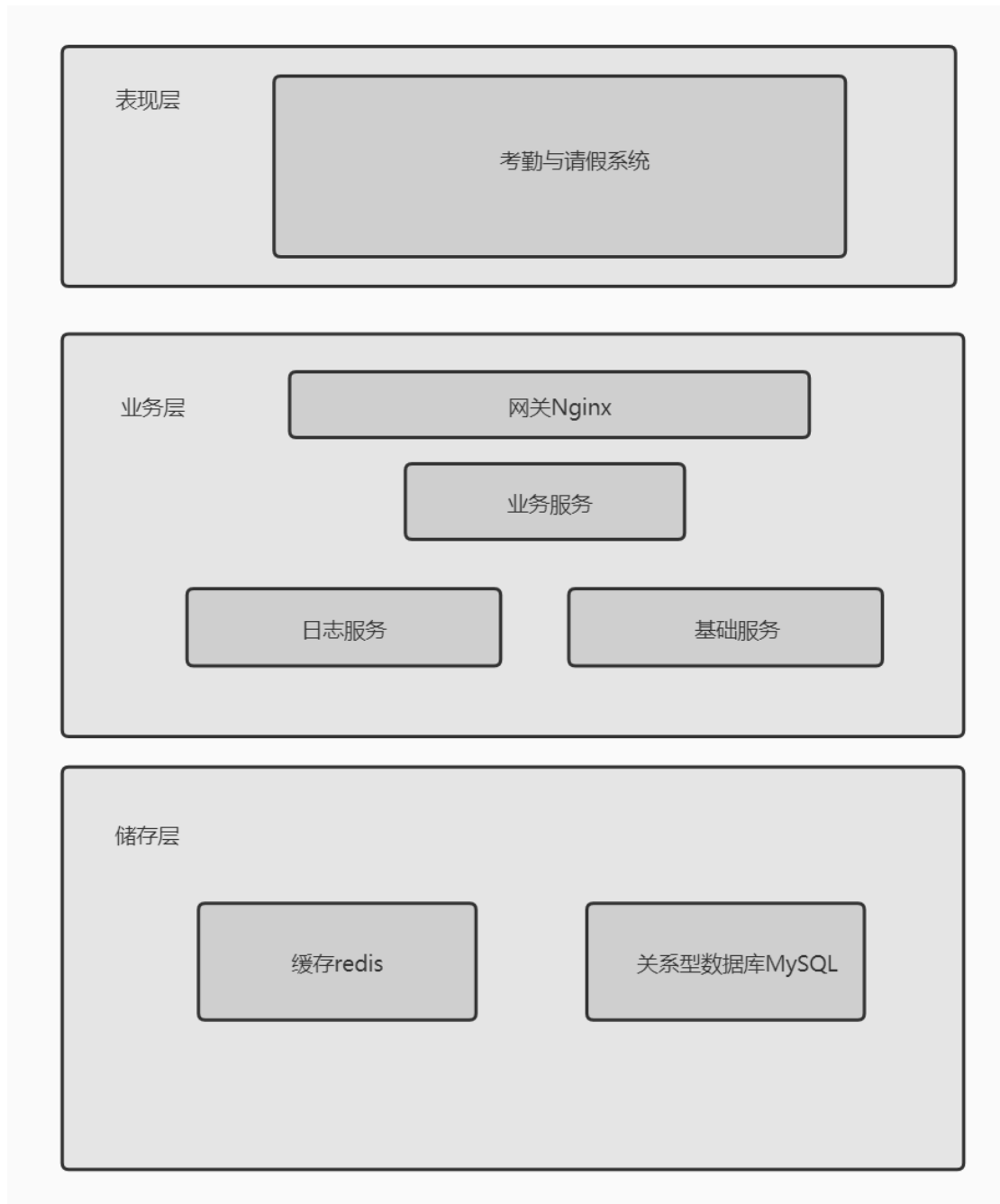
（2） 灵活部署、高效应用

考勤与请假系统具有模块级的灵活部署能力，可以方便地将开发的业务功能模块根据机构设置、业务配置、办理流程进行灵活部署，满足应用的要求。另外，系统具有灵活部署能力也使系统便于扩展升级。

（3） 先进实用，开发扩展

充分考虑到将来的技术发展需要，选择较先进、开放的技术与产品，采用国际或国内标准，建设具有一定先进性的网络和应用系统。网络及应用系统能满足实际应用的需要，而且维护方便、管理简单、使用灵活，产品和系统具有良好的性能价格比。降低对使用人员的计算机专业技术要求；功能必须符合和优化手工管理习惯，体现计算机管理的价值和优势，使工作人员能够方便快捷地共享、交流信息，高效地协同工作。

2.3 总体结构



系统总体结构图

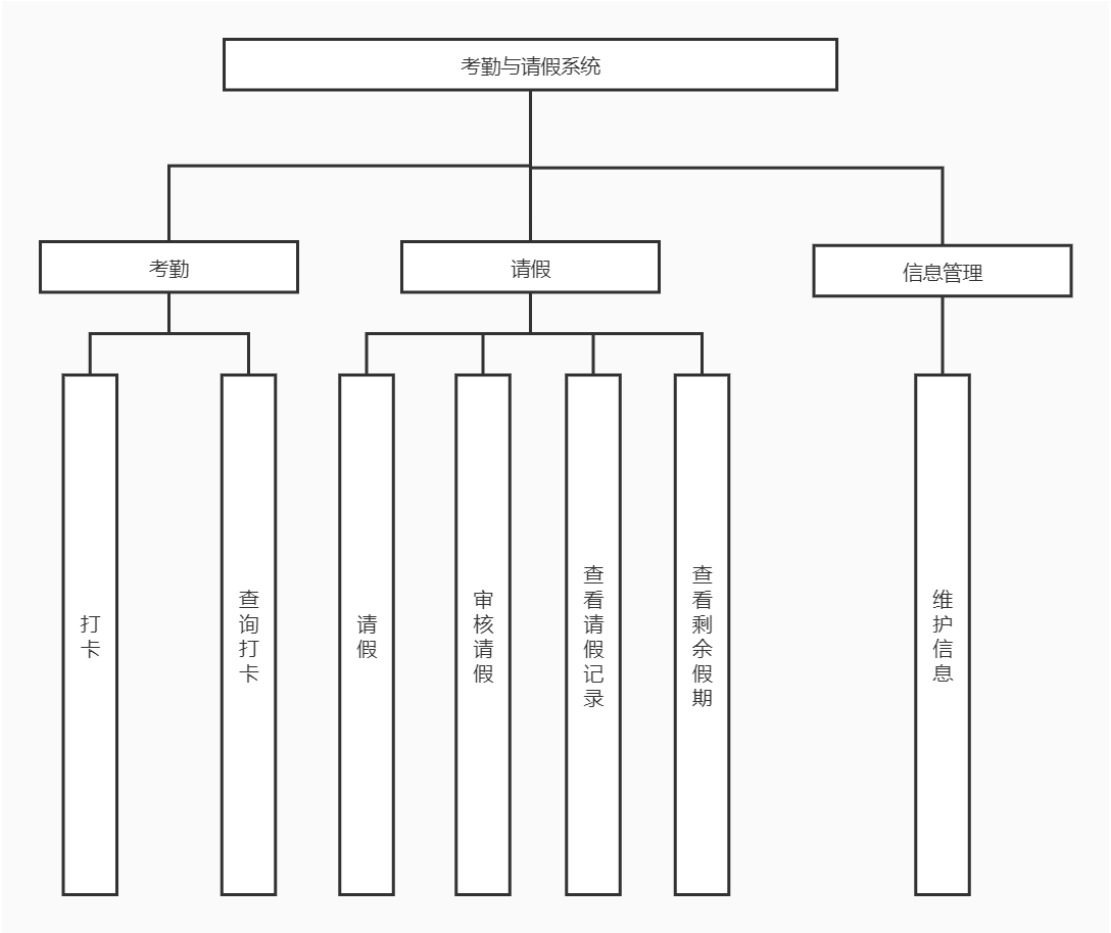
系统整体分为三层，表现层、业务层、存储层。

表现层：主要功能是显示数据和接受传输用户的数据，提供交互式操作界面，即社会治理指数评价系统。

业务层：处理用户的输入信息并保存，提供用户需要的数据；网关层处理用户请求并转发到各个服务，业务服务提供功能，基础服务为业务服务提供基础功

能，供业务服务调用。

存储层：存储用户输入的信息及平台产生的数据，并提供缓存功能。



系统功能结构图

考勤与请假系统，包括考勤、请假和信息管理三个模块，统筹解决部门员工的考勤和请假问题。

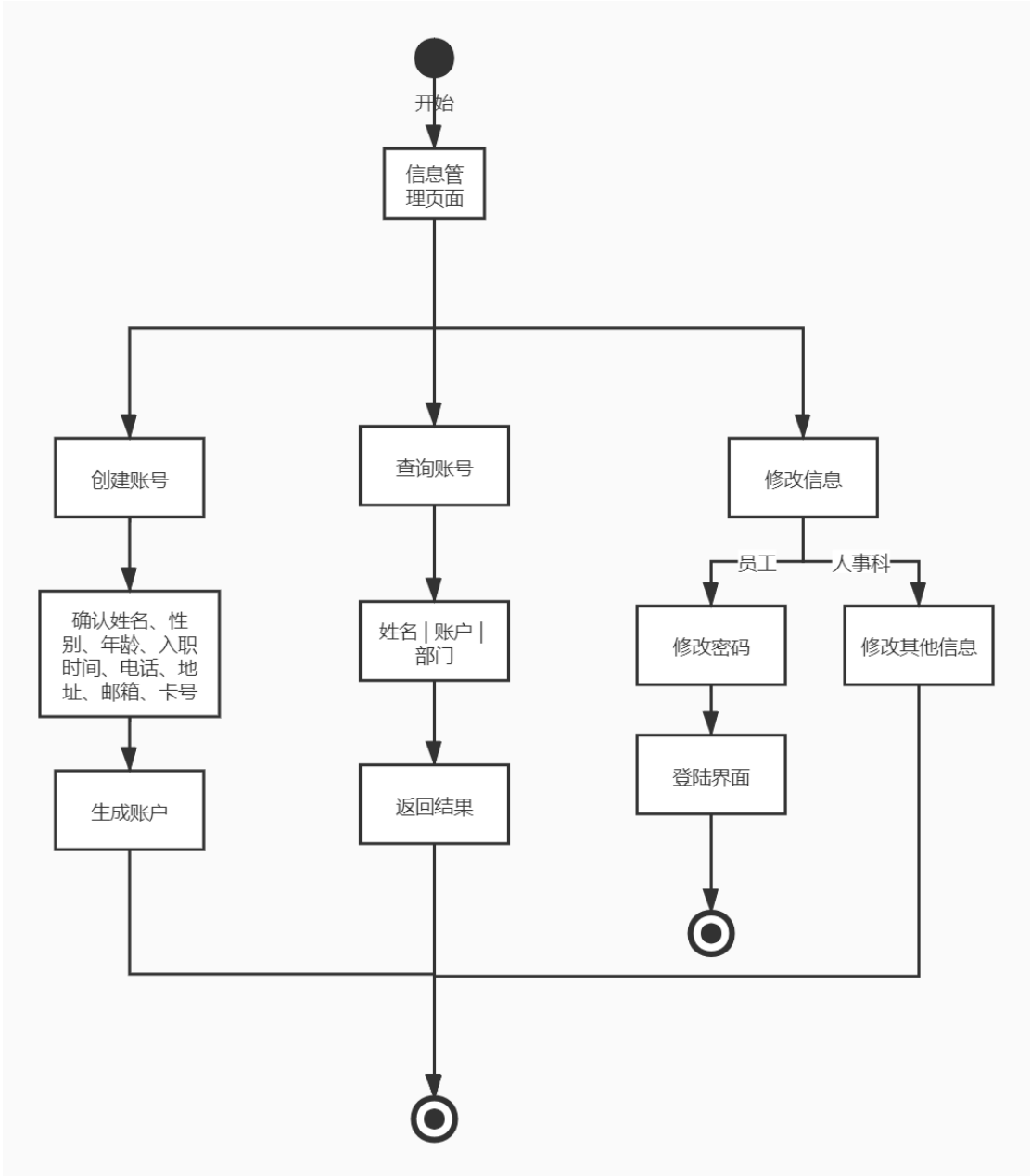
3.详细设计

3.1 信息管理

3.1.1 功能概述

普通员工需要提交个人信息材料到人事科处，人事科员工根据信息生成普通用户权限账号，并进行查询，人事科员工可以修改员工的所有信息，同时可以调节员工的权限，如升职后到人事科备注，人事科更改员工的权限。普通员工只可以自己的密码和账号

3.1.2 业务流程




3.1.3 创建账号

➤ 创建账号

● 功能说明


功能编号	1
功能说明	根据用户提交的信息为用户创建账户
输入数据	姓名、年龄、性别、入职时间、部门、电话、邮箱、地址、卡号

输出结果	显示账户和密码、权限等级
处理过程要求	账户不重复为 8 位数字串、权限等级初始为普通员工级别、密码设置为 8 位数字字母符合串
原型图	 <pre> graph TD A[创建账号] --> B[确认姓名、性别、年龄、入职时间、电话、地址、邮箱、卡号] B --> C[生成账户] </pre>
相关功能编号	

3.1.4 查询账号

➤ 查询账号

● 功能说明

功能编号	2
功能说明	根据部分信息查询员工
输入数据	姓名或者账户或者部门
输出结果	查询出来的用户具体信息
处理过程要求	可能查询出多条用户信息
原型图	 <pre> graph TD A[查询账号] --> B[姓名 账户 部门] B --> C[返回结果] </pre>
相关功能编号	

3.1.5 修改信息

➤ 修改信息

● 功能说明

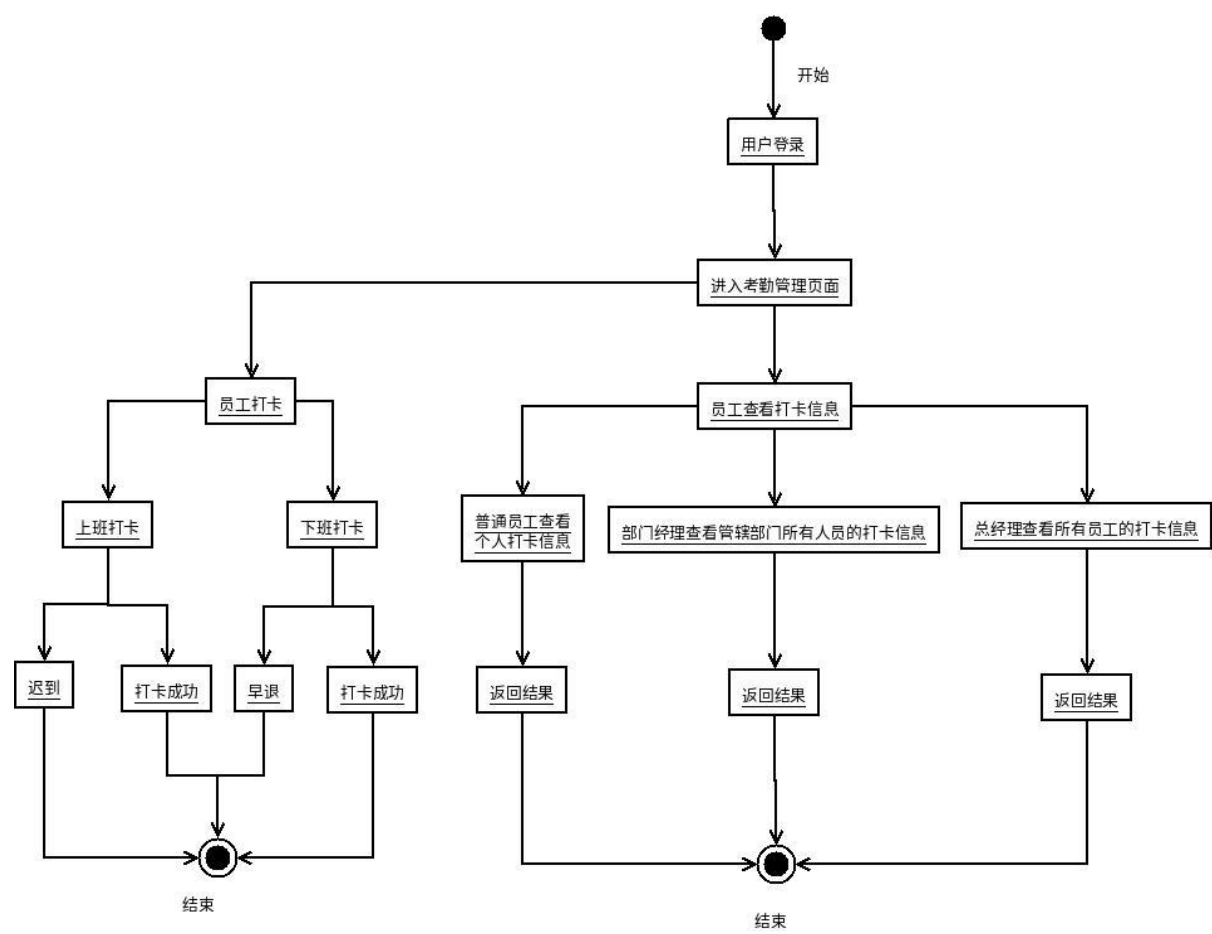
功能编号	3
功能说明	点击修改按钮，修改账户各种信息
输入数据	要求改的信息
输出结果	修改的结果
处理过程要求	人事科可修改所有信息，员工可以修改自己的账户和密码
原型图	<pre> graph TD Start(()) --> UC[修改信息] UC -- 员工 --> UC1[修改密码] UC -- 人事科 --> UC2[修改其他信息] UC1 --> UI[登陆界面] UI --> End((())) UC2 --> End </pre>
相关功能编号	

3.2 考勤管理

3.2.1 功能概述

普通员工上班打卡，下班打卡，可以查看打卡；部门经理可以查看自己部门员工的打卡记录、经理可以查看所有员工的打卡记录

3.2.2 业务流程

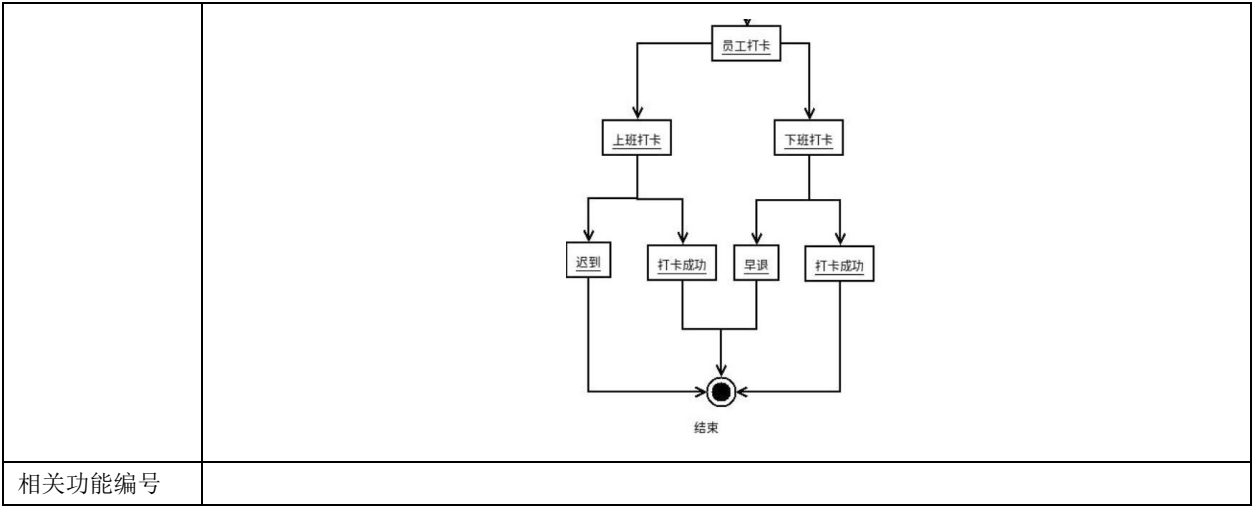


3.2.3 员工打卡

➤ 员工打卡

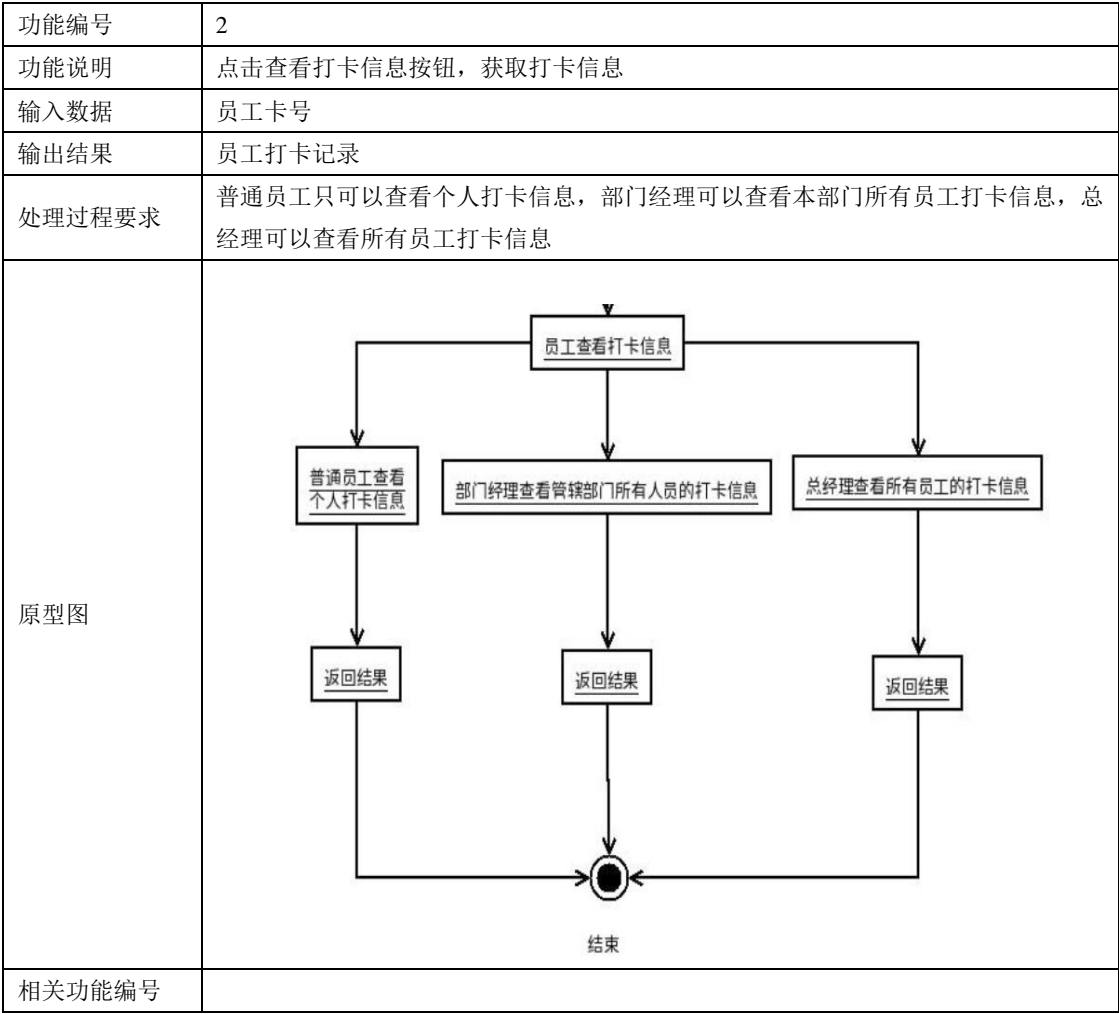
● 功能说明

功能编号	1
功能说明	点击打卡按钮，完成上班打卡或下班打卡
输入数据	员工卡号，打卡时间
输出结果	显示打卡是否成功，若失败，显示失败原因
处理过程要求	上班和下班各打卡一次
原型图	



3.2.4 员工查看打卡信息

- 员工查看打卡信息
- 功能说明

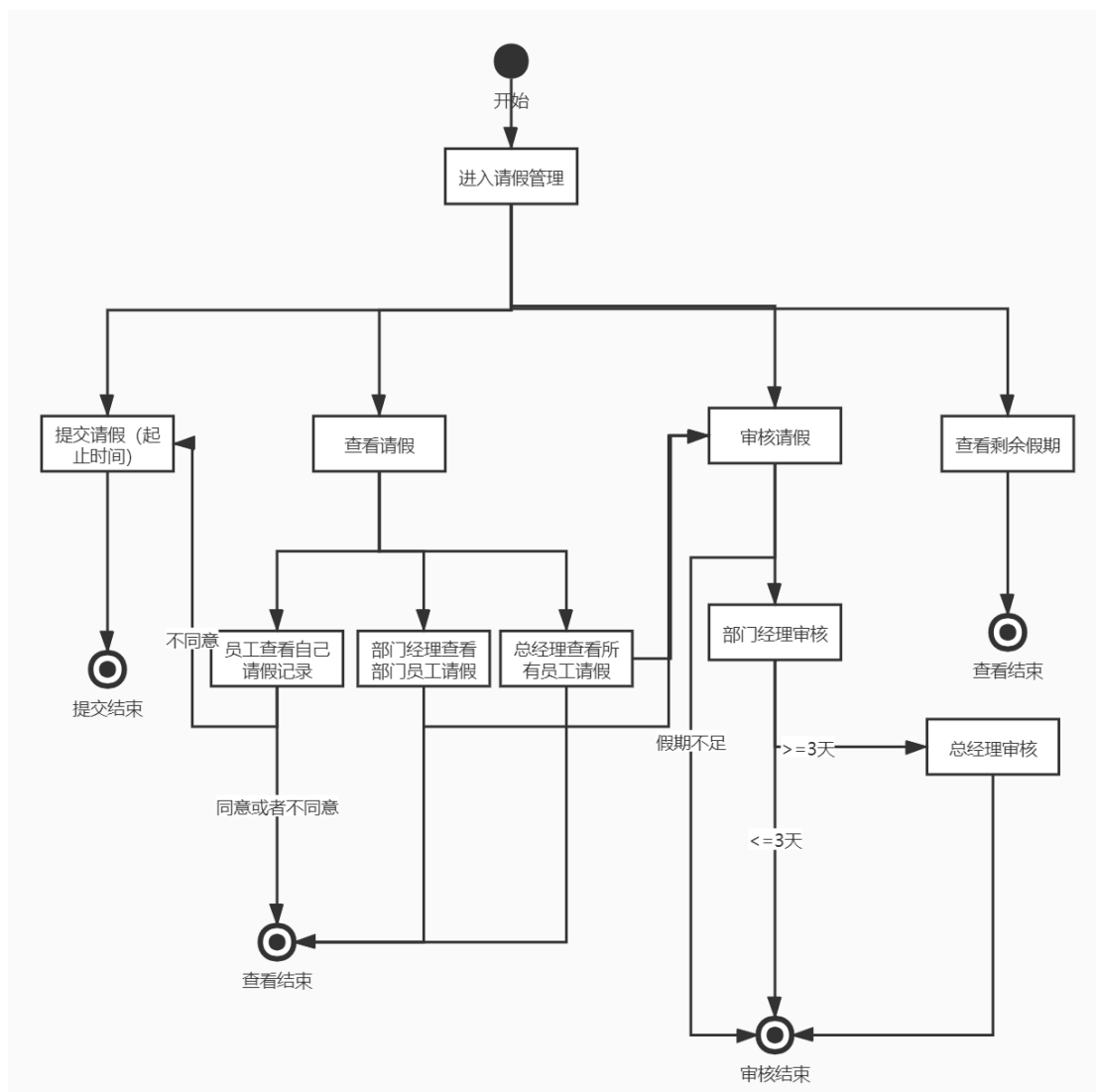


3.3 请假管理

3.3.1 功能概述

普通员工可以进行请假申请和查看申请，查看自己剩余假期。部门经理可以查看审核自己部门员工申请，生成任意时间范围内某人或本部门全体人员的请假记录。总经理可以查看审核所有员工请假申请，生成任意时间范围内某人或全体人员的请假记录。人事科人员可以在每月生成每位员工的报表。

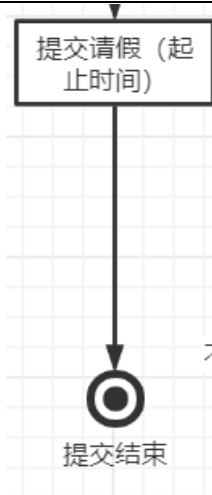
3.3.2 业务流程



3.3.3 提交请假

➤ 提交请假申请

● 功能说明

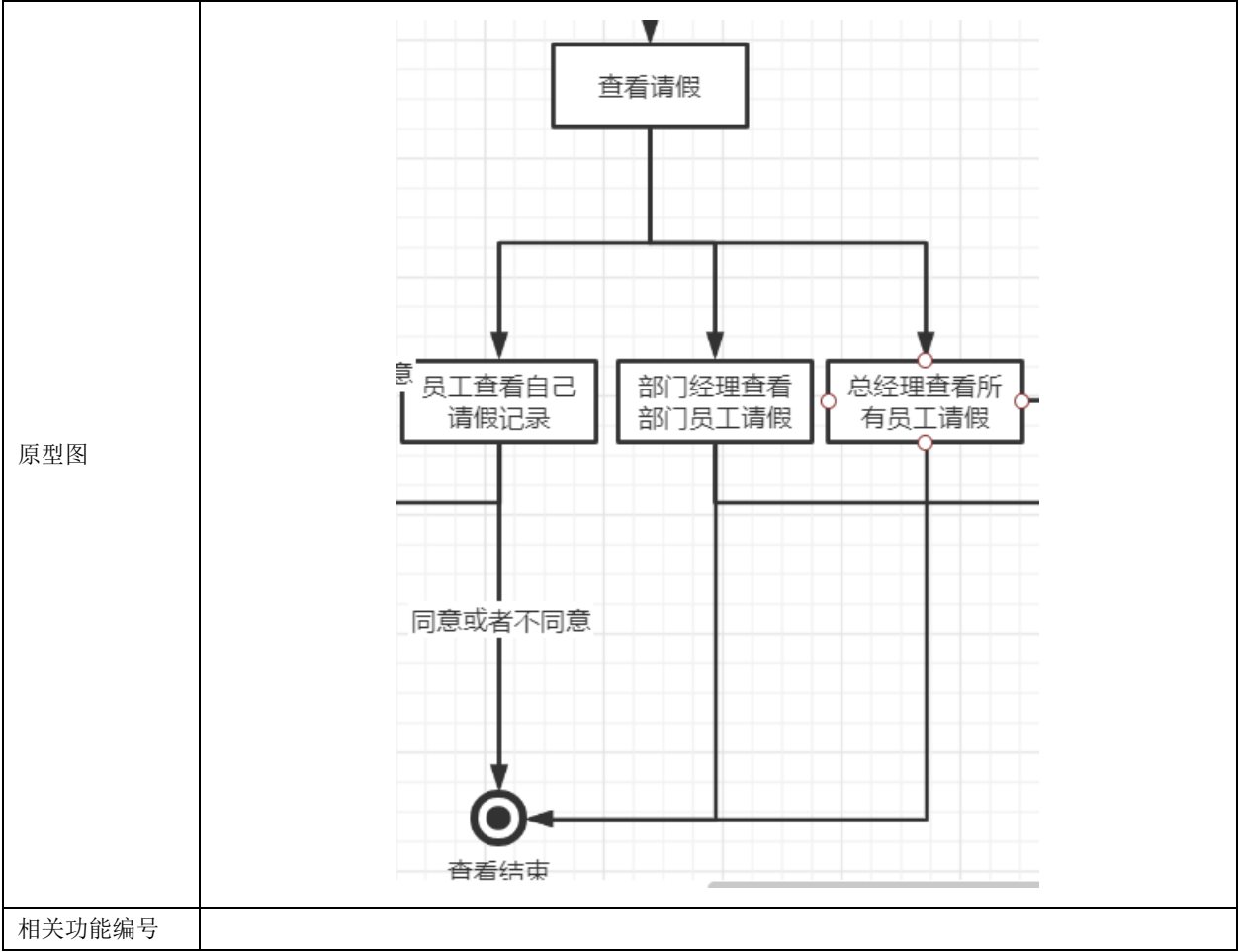
功能编号	1
功能说明	员工提交请假申请
输入数据	起止时间
输出结果	提交申请是否成功
处理过程要求	自动填充用户姓名、部门信息
原型图	
相关功能编号	

3.3.4 查看打卡

➤ 查看请假申请

● 功能说明

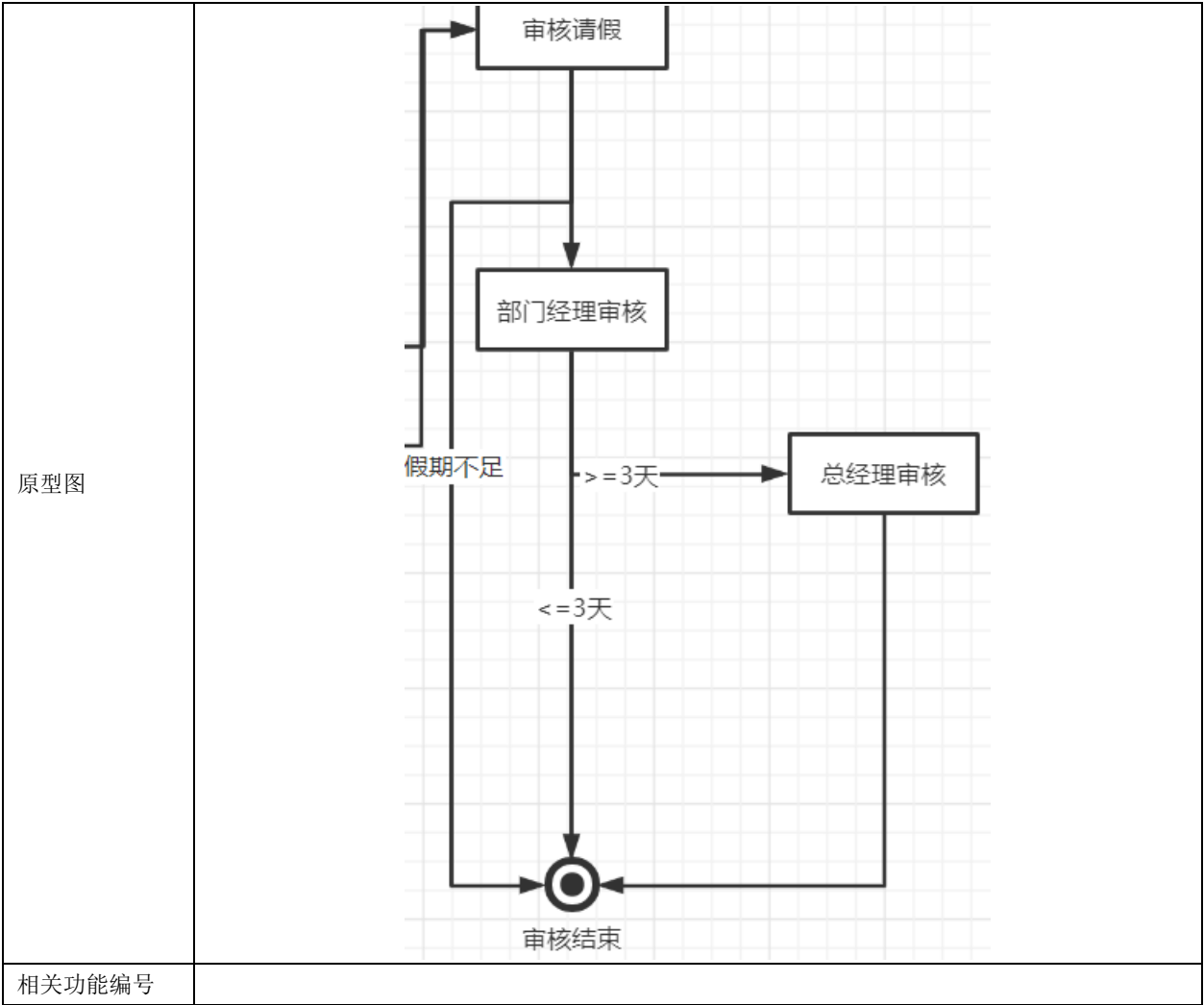
功能编号	2
功能说明	查看请假申请
输入数据	查询条件（员工：月（默认当月自己所有记录），部门经理：员工姓名（本部门）、月（默认当月自己部门所有记录），总经理：员工姓名、月（默认当月所有员工记录））
输出结果	请假申请记录
处理过程要求	不同用户权限展示不同功能



3.3.5 审核请假

- 审核请假
- 功能说明

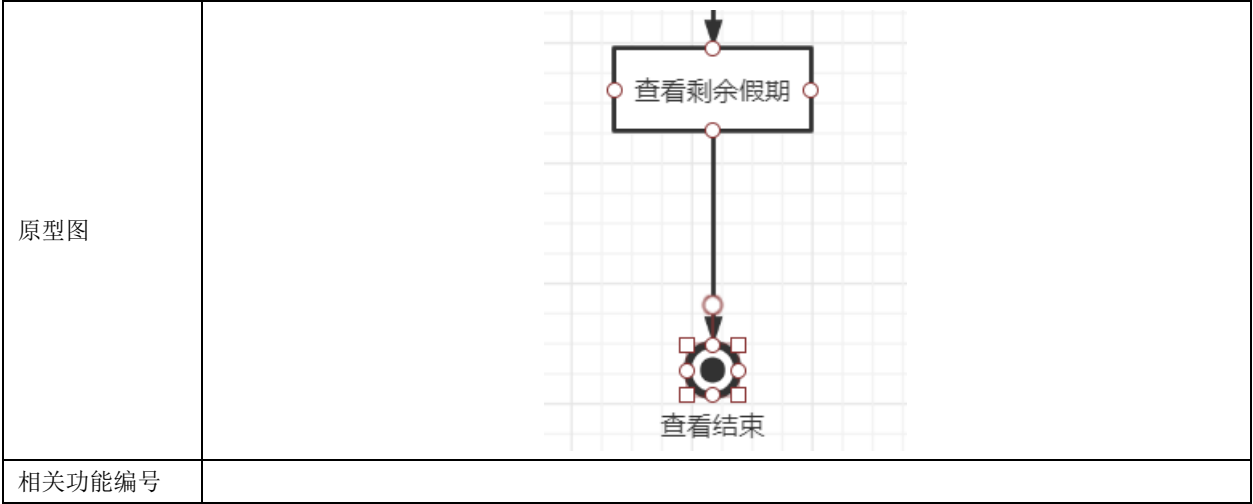
功能编号	3
功能说明	审核请假申请
输入数据	记录 id，审核意见，理由
输出结果	审核是否成功信息
处理过程要求	不同请假天数处理方式不同



3.3.6 查询剩余假期

- 查询基于假期
- 功能说明

功能编号	4
功能说明	查询剩余假期
输入数据	员工 id
输出结果	员工姓名，员工部门、剩余假期
处理过程要求	自动填充员工 id



4.接口设计

4.1 数据库设计规范

4.1.1 表设计规范

- 1、表名全部小写，单词间通过'_'间隔
- 2、主键命名为'id'，varchar(20)，不使用数据库的序列，应用生成全局序列。
- 3、必须包含 2 个审计字段且不能为空。created_time、updated_time。
- 4、关键词要求大写，使用 IDE 如 idea 进行格式化
- 5、常量枚举全部用大写

4.1.2 字段长度规则

名称类	类型	长度	备注
编码类	varchar	100	
账号类	varchar	100	如 email，username
状态类	varchar	5	如订单状态等
名称类	varchar	200	中文名称，如产品名

名称类	类型	长度	备注
手机电话	varchar	20	
描述简介	varchar	500	
网址类	varchar	500	如 url
时间类	datetime		

4.2 URL 和方法命名规范

4.2.1 RESTFUL URL 命名规范

API URI design API URI 设计最重要的一个原则： nouns (not verbs!) ， 名词（而不是动词）。

CRUD 简单 URI:

方法	URL	功能
GET	/user	获取用户列表
GET	/user/1	获取 id 为 1 的用户
POST	/user	创建一个用户
PUT	/user/1	替换 id 为 1 的用户
PATCH	/user/1	修改 id 为 1 的用户
DELETE	/user/1	删除 id 为 1 的用户

上面是对某一种资源进行操作的 URI，那如果是有关联的资源，或者称为级联的资源，该如何设计 URI 呢？比如某一用户下的产品：

方法	URL	功能
GET	/user/1/product	获取 id 为 1 用户下的产品列表
GET	/user/1/product/2	获取 id 为 1 用户下 id 为 2 的产品
POST	/user/1/product	在 id 为 1 用户下，创建一个产品

方法	URL	功能
PUT	/user/1/product/2	在 Id 为 1 用户下，替换 Id 为 2 的产品
PATCH	/user/1/product/2	修改 Id 为 1 的用户下 Id 为 2 的产品
DELETE	/user/1/product/2	删除 Id 为 1 的用户下 Id 为 2 的产品

4.3 方法命名规范

4.3.1 DAO

简单的 CRUD 请按如下规则命名

操作	例子	备注
增加	insert/add	
删除	delete	
修改	update	
查询	query	
搜索	search	

4.3.2 Service

简单的 CRUD 请按如下规则命名，其它操作请按业务动作命名，使用动词

操作	例子	备注
增加	add	
获取	get	获取到单条记录
删除	remove/delete	
更新	update	更新存在的记录
保存	save	更新，不存在则新增

操作	例子	备注
查询	query	根据 id 等简单条件查询
搜索	search	根据时间范围或模糊搜索

4.3.3 Rest

简单的 CRUD 请按如下规则命名，其它操作请按业务动作命名，使用动词

操作	例子	备注
增加	add	
保存	save	更新，不存在则新增
删除	remove/delete	
获取	get	获取到单条记录
更新	update	更新存在的记录
查询	query	根据 id 等简单条件查询
搜索	search	根据时间范围或模糊搜索

5.数据库设计

5.1 表结构

5.1.1 用户表（user）

属性	说明	类型	字段约束	字段类型	描述
Id	编码	Int	Not null	PK	
Did	部门 id	Int	Not null		
Dname	部门名称	Varchar	Not null		
Account	账号	varchar	Not null		
name	姓名	varchar	Not null		
gender	性别	Varchar	Not null		
age	年龄	Int	Not null		
Telephone	电话	Varchar	Not null		
e-mail	邮箱	varchar	Not null		
password	密码	varchar	Not null		
grade	用户级别	int	Not null		0: 普通员工 1: 部门经理

					2: 总经理 3: 人事科
--	--	--	--	--	------------------

5.1.2 考勤表 (user_attendance)

属性	说明	类型	字段约束	字段类型
Id	编号	Int	Not null	PK
uid	员工 id	Int	Not null	
Name	姓名	Varchar	Not null	
Did	部门 id	Int	Not null	
Dname	部门名	Varchar	Not null	
start_time	上班时间	Varchar	Not null	
end_time	下班时间	Varchar		
Length_absence	缺勤时长	Int		

5.1.3 请假表 (user_leave)

属性	说明	类型	字段约束	字段类型
Id	编号	Int	Not null	PK
uid	员工 id	Int	Not null	
Name	姓名	Varchar	Not null	
Did	部门 id	Int	Not null	
Dname	部门名	Varchar	Not null	
start_time	起始时间	Varchar	Not null	
end_time	结束时间	Varchar	Not null	
Length_leave	请假时长	Int	Not null	
Type	请假类型	Int	Not null	0: 事假 1: 年假
review_by_division_manager_	部门经理审核结果	Int		0: 审核不通过 1: 审核通过
Advice_by_division_manager	部门经理审核意见	Varchar		
review_by_division_manager_	总经理审核结果	Int		0: 审核不通过 1: 审核通过
Advice_by_division_manager	总经理审核意见	Varchar		
Result	最终结果	Int		0: 不通过 1: 通过

				2: 未审核 3: 部门经理 审核完
--	--	--	--	--------------------------

5.1.4 部门表（department）

属性	说明	类型	字段约束	字段类型
Id	编号	Int	Not null	PK
Name	部门名	Varchar	Not null	

6.环境配置

6.1 开发环境配置

说明本系统应当在什么样的环境下开发，有什么强制要求和建议。

表2. 开发环境

类别	标准配置	最低配置
计算机硬件		
软件		
网络通信		
其他		

6.2 运行环境配置

说明本系统应当在什么样的环境下运行，有什么强制要求和建议。

表3. 运行环境

类别	标准配置	最低配置
计算机硬件		
软件		
网络通信		
其他		

6.3 测试环境配置

说明本系统应当在什么样的环境下测试，有什么强制要求和建议。

- (1) 一般地，单元测试、集成测试环境与开发环境相同。
- (2) 一般地，系统测试、验收测试环境与运行环境相同或相似（更加严格）。

