# 绪论

## 研究背景

随着现代企业规模的扩大和工作效率的追求，企业员工考勤系统逐渐成为管理人员关注的焦点。传统的考勤方式，如手动记录和纸质签到表，存在诸多弊端，如易于造假、耗时耗力、数据管理困难等。为了提高考勤管理的准确性和效率，许多企业开始引入电子考勤系统。

企业员工考勤系统的研究背景主要源于以下几方面的需求。首先，随着企业规模的不断扩大，传统的考勤方式已无法满足大规模人员管理的需求，需要一种更高效的考勤系统来确保员工出勤的准确记录和统计。其次，企业对工作效率的要求不断提高，需要一种自动化的考勤系统，能够减少人力投入和准确计算员工的工时和加班情况。此外，员工的灵活工作模式和远程办公的普及，也对考勤系统提出了新的挑战和需求，需要适应多样化的工作场景和实时的考勤数据记录。

因此，企业员工考勤系统的研究旨在开发一种集成化的系统，能够实现员工的准确考勤记录、自动计算工时和加班情况、灵活适应多样化的工作模式，并提供可靠的考勤数据供管理人员参考和决策。这样的系统可以提高考勤管理的效率，减少人力成本，促进企业的正常运营和发展。通过对企业员工考勤系统的研究，可以为企业管理提供科学依据和技术支持，提升企业的竞争力和管理水平。

## 研究目的及意义

首先，研究的目的是提高考勤管理的准确性和效率。传统的考勤方式容易出现人为错误和数据丢失，而员工考勤系统可以自动记录员工的出勤情况，减少了手动操作的错误概率。同时，通过自动计算工时和加班情况，系统能够减少考勤统计的时间和工作量，提高管理人员的工作效率。

其次，研究的目的是实现灵活适应多样化的工作模式。随着员工的灵活工作安排和远程办公的普及，传统的考勤方式已不再适用。而网络形式的考勤则可以满足不同工作场景的需求，实现灵活的考勤管理。

此外，员工考勤系统的研究具有重要的意义。首先，它可以提供可靠的考勤数据和报表，为企业决策提供依据和参考。准确的考勤数据可以帮助企业掌握员工的出勤情况、加班情况和工时分布，为薪资发放、绩效评估和人力资源管理提供有力支持。其次，员工考勤系统可以减少人力成本，提高企业的运营效率。通过自动化的考勤管理，可以节省人力资源部门的工作量，使其能够更专注于其他重要的人力资源工作。

综上所述，企业员工考勤系统的研究目的在于提高考勤管理的准确性和效率，实现灵活适应多样化的工作模式，并具有重要的意义，为企业提供可靠的考勤数据和报表，减少人力成本，提高运营效率，为企业决策和人力资源管理提供科学支持。

# 可行性分析

## 技术可行性

本项目目标是开发一个基于C/S架构企业员工考勤系统，因此该系统包含两部分。桌面应用程序在已有的工作基础上进行二次构建，在无框架的环境下使用Core Java 进行开发。服务器同样在已有的工作基础上使用Microsoft SqlServer进行二次构建。因此本设计在技术要求上可行并且工作人员已有项目实践基础。主要的技术风险因素在于桌面应用程序没有使用框架搭建，因此部分功能实现相比于使用框架更加困难，以及UI美化，用户友好的优化难度更大。

## 经济可行性

本项目为Github开源公益项目，源码及开发文档均可免费下载。因此本设计不考虑系统出售所带来的经济效益。

成本方面，本组织目前成员1人，工作设备自带，开发环境为开源的VS Code以及SQL Server 2019 Developer（无经济效益），居家办公。因此成本可忽略不计。

## 社会可行性

项目中的引用均为开源代码、图片，项目本身不涉及任何侵权、妨碍等责任问题。再次重申本项目为开源公益项目，项目中所有对遵循 *MIT License* 规范被许可人的权利以及义务：

MIT License

Copyright (c) 2023 WakLouis

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## 操作可行性

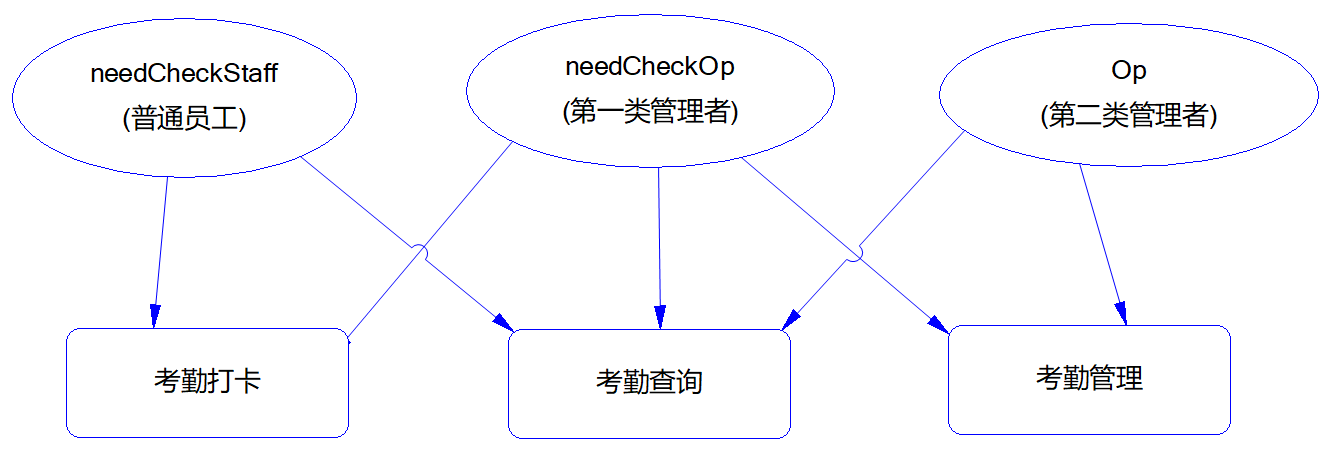
本项目以用户友好的角度制作桌面应用程序，将底层操作封装成功能提供给用户进行使用，即底层操作对用户透明，不需要用户进行学习即可使用本系统。系统可以提供直观易懂的用户界面和简便的操作流程，使员工能够轻松进行考勤记录和查询。

项目对公司的组织架构进行了抽象，从而增强了本系统的泛化适用能力，因此本系统在大部分的用户组织内都能够正常运行。

# 需求分析

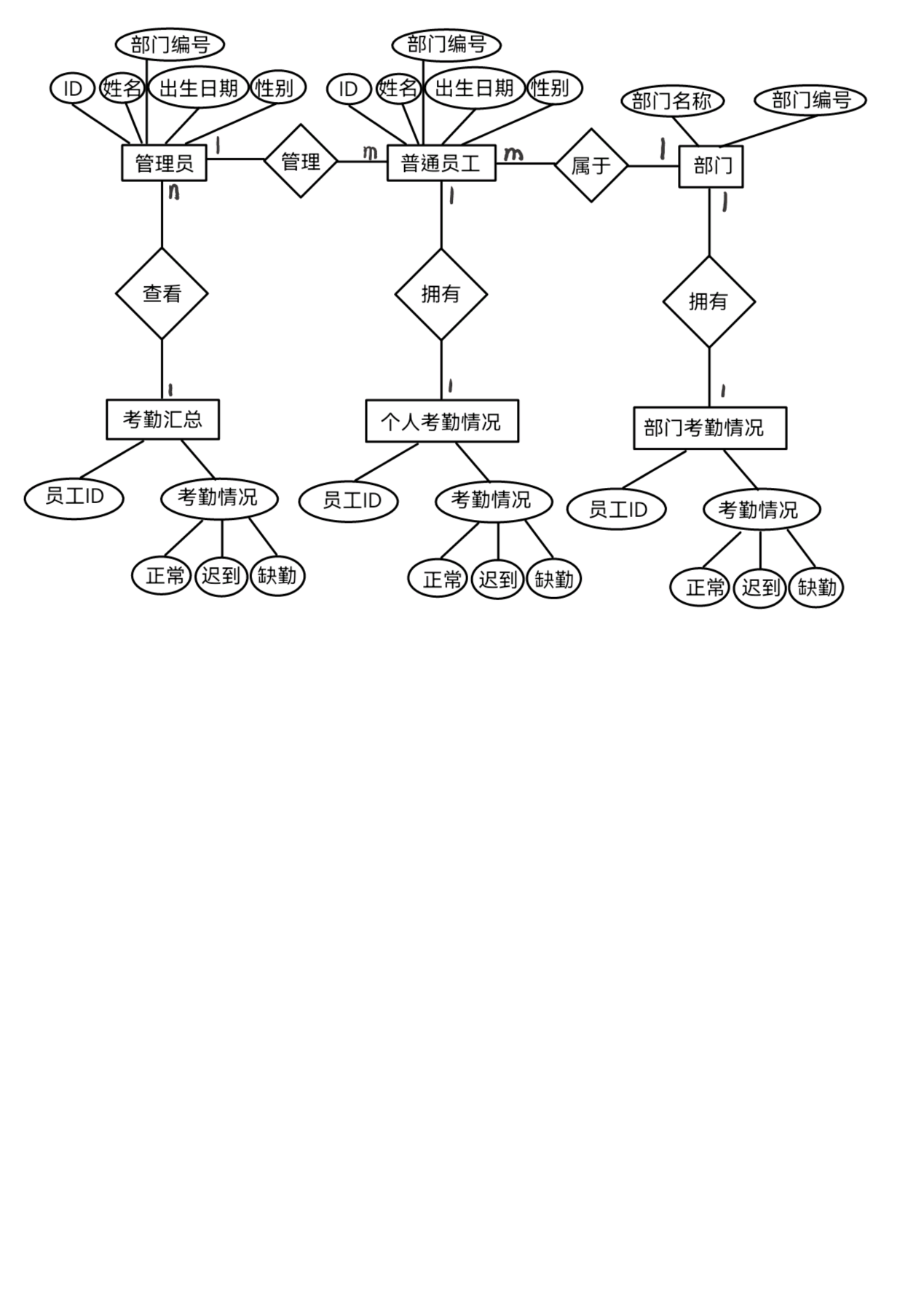
## 功能需求

首先考虑最基本的功能需求。使用本系统的用户身份至少分为两类：普通用户以及管理者，普通用户能够进行考勤打卡以及对自身考勤情况的查询；管理者能够查看或修改全体员工的考勤情况。本系统将管理者再次细分，分为一类管理员以及二类管理员，一类管理员本身不需要进行考勤打卡，并且能够查看或修改所有人的考勤情况；二类管理员本身需要进行考勤打卡，并且能够查看并修改除本身外所有人的考勤情况。此种权限划分能够对应于现实中的组织结构：老板和经理，老板不需要考勤打卡，并且拥有所有权限，经理需要考勤打卡并能够修改他人的考勤情况，但不能修改自身的考勤情况。下图为本系统的组织架构图。



除此之外，系统还将添加“站内邮件”功能，以便管理员与用户之间，用户与用户之间的交流互动。并且丰富了管理员的功能操作。

下图为本系统的E-R图，实体包含：考勤总表、员工表、部门表、邮件总表以及管理信息表。管理员和普通员工同时属于员工表，管理员管理普通员工，普通员工通过部门编号这一字段确定所属部门。管理员能够查看考勤总表，普通员工只能查看自身考勤情况，考勤总表以时间为自增字段，每条记录是员工编号，0代表缺勤，1代表正常，2代表迟到。



E-R图

## 性能需求

本系统主要影响性能的因素在于对并发需求的响应，即性能与员工人数基本成负相关，即同时使用人数越多，性能越差。但是现实生活中，考勤系统并不会在短时间内收到极高的用户指令，因此系统的并发响应能力对性能因素并不会造成太大影响。

其次本系统主要针对于中小型企业日常考勤需要，其员工人数十几人至上百人不等，最高不超过千人，因此对于服务器来说完全足以应付这种规模的并发响应。

因此本项目在如此环境下不太需要深究其性能需求。

## 可靠性和可用性需求

本系统功能设计并不复杂，数据流条理清晰，每一个分支都有其对应的处理操作，基本可以做到无故障持续运行，因此本系统可靠性相对来说比较优秀。

由于客户端必须连接服务器，因此可用性取决于服务器的正常运行，而本系统没有需要服务器进行停机维护的操作，因此可用性相对来说也比较优秀。

## 出错处理需求

本系统对于所有用户的输入问题以及网络连接问题，都有相对应的报错信息，当问题产生时，系统将会将相应的错误提示以文本的方式反馈给用户。

## 接口需求

* + 1. boolean connection(String user, String paswd, String ipaddress)

用于进行客户端与服务器之间的连接测试。user为登录用户名，paswd为登录密码，ipaddress为服务器地址。成功返回true，失败返回false。

* + 1. boolean fCheck()

用于进行考勤打卡。使用当前登录用户作为打卡用户，成功打卡返回true，失败返回false。

* + 1. void fQuery()

用于查询自身考勤情况。使用当前登录用户作为查询用户，将查询结果打印于显示窗口，无返回值。

* + 1. void fQueryAll()

用于查询所有用户的考勤情况，将查询结果打印于显示窗口，无返回值。

* + 1. void fUpdate(String Name, String Date, String To)

用于管理员更新某一用户的考勤情况。Name为被更新用户ID，Date为更新日期，To为更新值。操作成功返回true，失败返回false。

* + 1. ArrayList<Mail> fGetMail()

用于获取所有发给自己的邮件。使用当前登录用户作为查询用户，返回Mail类的动态列表，无邮件时返回null。

* + 1. boolean fSendMail(String receiverIDSet, String title, String content)

用于用户发送邮件给其他用户。receiverIDSet为接收者集合，以逗号分隔，title为邮件标题，content为邮件内容。发送成功返回true，失败返回false。

* + 1. boolean fCreateNewUser(String userID, String departmentID, String name, String sex, String birthday, String paswdForCreate)

用于管理员创建新用户。userID为新用户ID，departmentID为所属部门ID，name为新用户姓名，sex为新用户性别，birthday为新用户出生日期，paswdForCreate为新用户登录密码。创建成功返回true，创建失败返回false