

成绩：

评语：

指导教师：

年 月 日

# 西南大学

## 计算机与信息科学学院

### 课程设计报告

课 程： 基于 Java 平台的 Web 应用开发

题 目： 基于 java 开发的 web 项目：人员管理系统

GIT 地址： <https://github.com/mingzhixian/study.git>

级、专业： 2020 级 网络工程 专业 2 班

学生姓名： 郭晓瑞

提交日期： 2022 年 1 月 8 日

## 目 录

目 录.....	II
第一章 绪论.....	3
1.1 系统的目的和意义.....	3
第二章 人员管理系统需求分析.....	3
2.1 引言.....	3
2.2 需求分析.....	3
2.2.1 数据存储.....	3
2.2.2 用户登录与注册.....	4
2.2.3 防止入侵.....	4
2.2.4 效率优化.....	4
2.2.5 简化部署.....	4
第三章 人员管理系统结构分析与设计.....	4
3.1 引言.....	4
3.2 系统模块结构图.....	5
3.3 系统各模块分析.....	5
3.3.1 登录与注册模块.....	5
3.3.2 增删改查模块.....	5
3.3.3 权限控制与缓存管理模块.....	6
3.3.4 前端界面与动态更新模块.....	6
3.3.5 数据库管理模块.....	7
第四章 关键技术研究.....	7
4.1 实现技术路线.....	7
4.2 关键技术研究.....	7
4.2.1 瀑布流效果.....	7
4.2.2 项目初始化.....	8
4.2.3 管理用户在线.....	8
4.2.4 数据库连接池.....	8
4.2.5 Ant 配置.....	8
4.2.6 文件上传与下载.....	8
第五章 系统实现.....	9
5.1 系统实现介绍.....	9
5.2 系统实现的不足.....	12

---

# 第一章 绪论

## 1.1 系统的目的和意义

本项目的目标人群是对一些有人员管理需求，但功能需求并不大的构或单位，如寝室楼人员记录管理。

在寝室人员记录管理时，因人员较多，信息量较大，传统手写笔记或依靠管理员记忆的方法显然并不合适，其次尽管 excel 软件可以实现记录统计所有人员的功能，但是该软件功能过多，使用较为复杂，其中功能有很多是管理员并不需要的，这造成管理员的学习成本增加，无法短时间内迅速完成软件使用方法的学习，同时，因 excel 软件功能太多，软件体积较大，对设备运行要求较高，这会增大管理员的使用成本，且 excel 类软件电脑端与手机端软件相差较大，手机端软件的功能性稍弱，操作逻辑与电脑端不同，这又会增加学习和使用成本，因此为解决此问题而诞生了本项目。

本项目实现的功能是对人员信息的存储与查询，相比较使用 excel 软件处理表格进行存储信息的方式，本项目具有允许多用户操作、允许上传下载附件、兼容性广、操作简便易懂的优势，虽然 excel 软件提出了云办公的方式，可以解决多人同时操作的需求，但此方式对性能要求较高，而本项目因基于网页，大部分设备（如手机终端、个人电脑）均可以使用，无需安装额外软件，占用小，同时本项目实现了 excel 没有的附件功能，更符合用户使用需求，且因本项目操作基于网页，兼容性较广，Linux、Android、Windows、MacOs、Ios 系统均可操作，且界面几乎相同，这进一步降低了学习使用成本，此外本项目实现了权限控制，如限制用户只能搜索查看，不允许修改删除添加等操作，更符合用户需求。

## 第二章 人员管理系统需求分析

### 2.1 引言

本项目的目的是实现人员信息的存储以及简单处理，并尽可能简化操作步骤。

本项目的主要内容是使用网页实现人员信息的存储、添加、查询、修改、删除功能，实现用户权限控制。

本项目实现的手段是依托数据库进行存储人员信息，并依托 java 作为网站后端，以网页的形式实现对数据库信息的查询修改添加删除等一系列操作。

### 2.2 需求分析

#### 2.2.1 数据存储

本项目首要需求是对人员信息的存储，本项目采用了数据库进行存储，因项目较为简单，

---

为了项目部署步骤简化以及节省服务器压力，采用了 sqlite 数据库，以 PeoManager.db 为数据库，以 peo 表存储人员信息，以 count 表存储网站用户信息，通过 java 编写的 servlet 实现对数据库的访问与操作。

## 2.2.2 用户登录与注册

为防止信息被随意改动，需设置用户权限，不允许非授权用户对信息进行添加、删除、修改等操作，因此增加了用户注册与登录功能，访问项目均需登录，且注册时默认无修改、添加、删除功能，但可由拥有权限的用户授与管理员权限，其中为进一步提高安全性，授权界面的链接并不会出现在项目正常使用界面中，隐藏授权界面只能通过直接输入网址访问，且访问时会检查当前用户是否为管理员。

## 2.2.3 防止入侵

对于一个管理系统，应以稳定安全为先，因此访问系统任何界面均需要进行登录，且登录界面设置验证码防止暴力测试密码。

## 2.2.4 效率优化

对于一个在线信息管理系统，需要高效率，因此本项目需要对数据库设立连接池，减少频繁与数据库建立链接、销毁链接，同时需要优化静态文件缓存，保证样式表以及 js 等文件及时更新的同时尽可能减少带宽占用，以及对传输文件进行压缩，减少换行符、空格符等无用字符。

## 2.2.5 简化部署

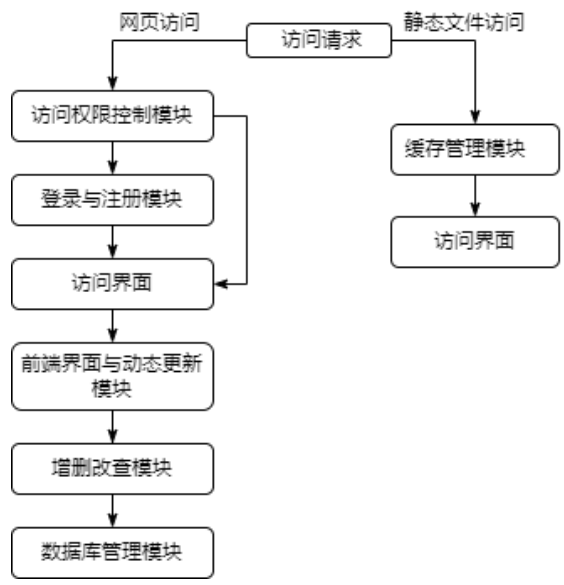
对于项目来说，为提高项目易用性，应简化部署的困难程度，因此项目采用了简单易安装的 sqlite,且项目实现了自动初始化数据库的建立、表的建立、以及初始测试用户的建立等功能，避免用户在部署时接触数据库建立等复杂操作，减少项目部署使用难度。

# 第三章 人员管理系统结构分析与设计

## 3.1 引言

本项目设计了登录与注册模块、增删改查模块、权限控制与缓存管理模块、前端界面与动态更新模块、数据库管理模块共五个模块，以下为这些模块的具体功能和实现方法思路，以及五个模块之间的连接关系。

## 3.2 系统模块结构图



## 3.3 系统各模块分析

### 3.3.1 登录与注册模块

在访问任何界面之前均会由权限控制模块拦截至登录与注册界面，此模块对应 login servlet,用户信息存储在 PeoManager 数据库的 count 表中，每一个用户拥有三个属性，分别为账户名、密码、admin，其中账户名不允许重复，admin 为权限控制模块检查当前用户是否具有修改删除添加权限提供支持，默认新注册用户 admin 应为 0。

Login servlet 的 get 方法提供登录、登出、注册界面，其 post 方法提供对用户登录、登出、注册行为的处理（包括处理数据库信息、管理 cookie）。

get 方法在无登录标记 cookie 时返回登录界面（登录与注册界面相同，具体由 post 方法区分），当有登录标记 cookie 时返回登出界面。

post 方法在无登录标记 cookie 时优先检查验证码是否正确，不正确返回错误信息，若验证码正确则获取提交表单中用户名，根据用户名查找数据库 count 表，若查询为空则视为注册新用户，添加数据库新用户信息后返回提醒用户注册成功；若查询不为空则检查密码是否正确，根据结果返回登录成功或失败，若成功则在返回头中添加登录标记 cookie。若 post 请求中存在登录标记 cookie,则将请求视为登出操作，将登录标记 cookie 消除（过期时长设为 0）。

### 3.3.2 增删改查模块

增删改查模块依托数据库管理模块，添加为 Add servlet，删除与修改为 Update servlet，查找为 Find servlet 与 All servlet。

Add servlet 的 get 方法提供添加界面，post 方法获取提交的表单，并检查是否存在 id 重

---

复的错误，检查无误后组合相应 sql 语句并传递至数据库管理模块。

Update servlet 的 get 方法提供人员修改界面，此界面可由查找界面和所有人员界面进入，其 post 方法对用户的修改与查询请求进行处理，根据用户点击不同按钮由前端界面与动态更新模块自动在提交的表单中注入用户请求操作以及原 id 号，post 方法根据此进入不同处理，若为删除请求则直接组合相应 sql 语句并传递至数据库管理模块进行删除操作，若为修改请求则检查用户是否修改了 id，若有修改则检查是否造成冲突，无冲突或未修改 id 则根据原 id 号组合 sql 语句并传递至数据库管理模块进行修改操作。

Find servlet 与 All servlet 相似，Find servlet 的 get 方法提供查找界面，post 方法依据请求的表单生成对应 sql 语句并传递至数据库管理模块，根据数据库返回值生成查找结果。当查找值为空时则默认为查找所有与 All servlet 的 get 方法类似。其中查询界面与所有记录界面每次均只展示 10 条记录，当 10 条记录均浏览完成后由前端界面与动态更新模块发送请求更新下 10 条记录，以达到瀑布流的效果，将数据处理压力分散为多次。

### 3.3.3 权限控制与缓存管理模块

权限控制模块依托 Filter servlet，负责设置各 servlet 的编码和拦截未登录用户访问以及拦截无权限用户访问添加、修改、删除、权限管理界面。权限管理界面为 Uset servlet，负责对当前注册用户是否拥有管理员权限进行管理。缓存管理模块依托 Cache servlet，负责设置静态文件如 css、js、img、字体等的缓存时间。

Filter servlet 会先设置 HttpServletRequest 与 HttpServletResponse 的编码格式，统一为 UTF-8 编码，其后会检测请求中是否存在登录标记 cookie，若不存在则会拦截至登录界面，若存在则会进一步检测是否为添加、删除、修改以及权限管理界面以及当前用户是否有访问权限，若无会拦截请求并返回权限不足提示。

Cache servlet 负责设置静态文件的缓存时间，其中字体文件的缓存时间为 100 天，其余则为 1 天，保证及时更新样式文件的同时减少每次访问时间。

Uset servlet 的 get 方法根据数据库管理模块查询当前所有注册用户，并现实其是否具有管理员权限，返回管理网页。其 post 方法根据提交的表单组合相应 sql 语句，更新注册用户管理员权限。出于安全考虑，此界面隐藏，项目主界面无本网页链接，需从网址进入，且会通过权限控制模块检测，只允许管理员操作。

### 3.3.4 前端界面与动态更新模块

本模块分为两部分，前端代码提供模块为 GetHtml.java，其将网站中个界面的各元素分离，构建模板，供各类小组件，如网页头部、底部、添加界面的 Form 等等，并提供 GetHtmlSelect 函数实现调用上述组件，在 GetHtmlSelect 返回相应值之前会调用 ZipHtml 函数依托 HtmlCompressor 对网页代码进行压缩，减少空格、换行符。

动态更新模块主要为 java script，包括依托 anime 框架搭建的多种动画，以及依托 jquery 中 ajax 功能实现的触底动态加载新内容实现的瀑布流效果，依托 canvas 实现的网页线条背景等等。

---

### 3.3.5 数据库管理模块

数据库管理模块为 Dbtool java,其提供数据库初始化、对数据库执行 sql 语句、根据相应 sql 语句返回相应对象数组的功能。

数据库管理模块在部署完成后初次运行时会检测是否存在数据库文件,若不存在则会自动建立数据库文件,并建立人员信息标 Peo 以及注册用户标 count,并生成测试数据帮助使用者测试使用。(初始化数据库地址由 Dbtool.java 文件下 dataPath 变量指定,另项目人员管理上传的附件等数据文件也存放于 dataPath 下,将数据与项目分离,便于更新项目。)

本模块依托 HikariCP 实现数据库连接池,避免数据库频繁建立、销毁连接,提供执行单条 sql 语句的 excute 方法与执行多条语句的 excuteBatch 方法、查询表记录数目的 allEle 方法、根据 sql 语句查询所有人员信息记录的 all 方法、以及根据 sql 语句查询所有注册用户信息记录的 login 方法、检测 id 是否冲突的 ishave 方法。

## 第四章 关键技术研究

### 4.1 实现技术路线

因项目初衷并非大项目,追求简单高效,因此数据库选择简单易懂、占用小、大多 linux 系统自带的 splite,减少部署难度,减少项目占用。

因 java 在 web 项目搭建方面较为方便,所以使用 java 作为项目语言。

因在项目规划中设计了项目网页模块化、统一化,所以网页表现较为简单,因此不使用注重表现的 jsp,只使用 servlet 进行网站搭建。

因项目追求简单,且前面提到网页的模块化、风格统一化,所以未使用 bootstrap 等快速建站工具,使用自编网页并增加部分动画实现客户端界面。

网页前端引入了 anime 框架,因为 anime 较为轻量,带宽占用小,且动画不复杂,对浏览器压力小。

数据库连接池工具采用 HikariCP,因为根据某研究报告,其速度远超 BoneCp,且 Spring Boot 将在 2.0 版本中把 HikariCP 作为其默认的 JDBC 连接池。

前端 js 引入 Jquery,因为其提供的功能十分全面。使用其中 ajax 功能进行异步刷新实现了瀑布流效果。

### 4.2 关键技术研究

#### 4.2.1 瀑布流效果

瀑布流效果主要依托触底监听以及 ajax 异步加载并刷新,其中监听器较为耗费资源,且刷新次数控制不好会影响画面,因此使用 rAF (requestAnimationFrame) 触发滚动事件,每次浏览器刷新界面时检测是否到达底部,其次为减少用户等待时间,应改检测是否到达底部为距离底部小于 80px,提前触发异步加载。其次因在底部附近滚动易多次触发异步加载函数,为减少服务器负担,采用了防抖函数,将多次触动集中为一次,减少客户端与服务器

---

端压力，最后因为数据量有限，为防止已展示所有数据后仍不断向服务器请求数据，设计了在加载完所有数据后停止监听触底加载，减少压力。

## 4.2.2 项目初始化

为减少项目部署难度，需尽可能减少数据库等相关操作，避免用户手动配置相关环境，因此项目编写了数据库初始化的脚本，并在项目部署后自动检查是否拥有数据文件，若无则自动建立相关文件夹并依据 `jdbc.ScriptRunner` 运行数据库脚本初始化数据库。

## 4.2.3 管理用户在线

项目中使用 `cookie` 记录当前登录用户，`cookie` 存在一定过期时间（20 分钟），因每次访问网页均会经过 `Filter servlet`，在拦截器中会刷新 `cookie` 过期时间，即当用户距离上次操作网页时间少于 20 分钟时，无需再次登录，无论中途是否关闭当前网站或浏览器。

其次项目使用 `session` 监听当前访问人数，当 `session` 建立时计数增加，当 `session` 关闭时计数减少，当前访问人数可通过点击任意网页的表头进入 `Index` 界面查看。

## 4.2.4 数据库连接池

项目使用了数据库连接池来避免频繁建立、销毁连接，实现连接的高效、安全的复用，在比较各类数据库连接池工具后，最终选用的是 `HikariCP`，处于项目整体考虑，将连接池最大连接数设置为 10，数据库连接池。

## 4.2.5 Ant 配置

项目由 `idea` 编写，对于使用 `idea` 软件进行修改有着强大的兼容性，且项目提供 `ant build.xml` 文件，在代码修改后可直接在项目根目录运行 `ant` 命令打包为 `war` 文件。（`build.xml` 文件由 `idea` 插件 `Generate Ant Build` 自动生成）。

## 4.2.6 文件上传与下载

项目设计了附件的上传与下载功能，考虑到项目版本更新的方便性，附件文件需与项目隔离开，因此项目的附件与数据库一起存放在项目外某一地址，因此需设计文件下载 `servlet` 以实现访问项目根目录以外的文件，其次考虑到用户上传的附件可能存在重名现象，因此在文件上传时应进行重命名，且为了方便用户，下载时需要重新命名至上传时的文件名。



# 第五章 系统实现

## 5.1 系统实现介绍

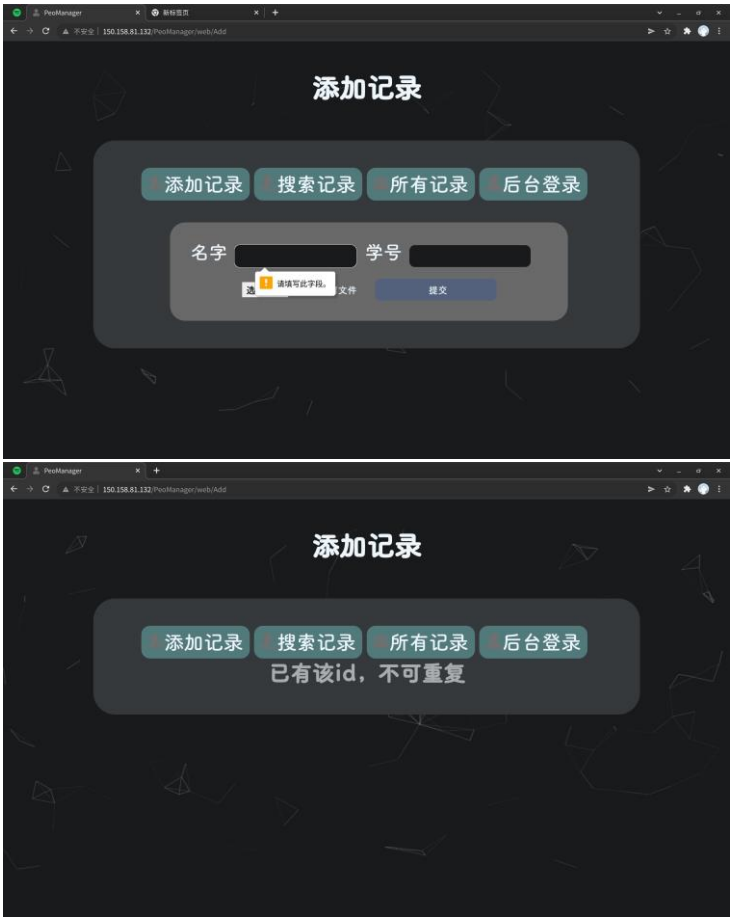
系统设置了拦截器，访问任何界面均需先进行登录（登录界面会判断是否为空，验证码不区分大小写，可通过点击验证码更新），若无此用户会自动注册新用户，新用户注册完成后默认无修改删除添加权限（默认唯一拥有所有权限的用户为 admin,密码为 admin,登录拥有管理员权限的用户后访问网站 url 加/web/uset 网页可对现有用户的权限进行控制）：



在进入系统后，可通过上方四个选项进入相应功能，其中无权限用户在进入添加和修改删除、权限管理界面时会提示权限不足：



添加界面会检测是否为空，且 id 字段只允许输入数字，提交后会先检测是否 id 冲突：

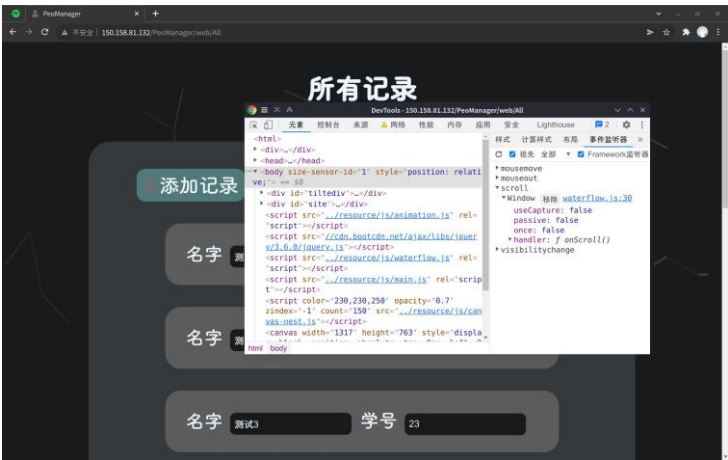


查找界面可单独限制姓名、单独限制 id 也可姓名 id 均作限制，均不做限制时默认为查询所有记录，从查找结果或所有记录界面单击某一条记录可进入详情界面，在该界面可下载附件（若无附件则不显示下载按钮），也可进行姓名与 id 的修改，修改时会检测当前用户是否为管理员，同时检测修改是否产生 id 冲突：

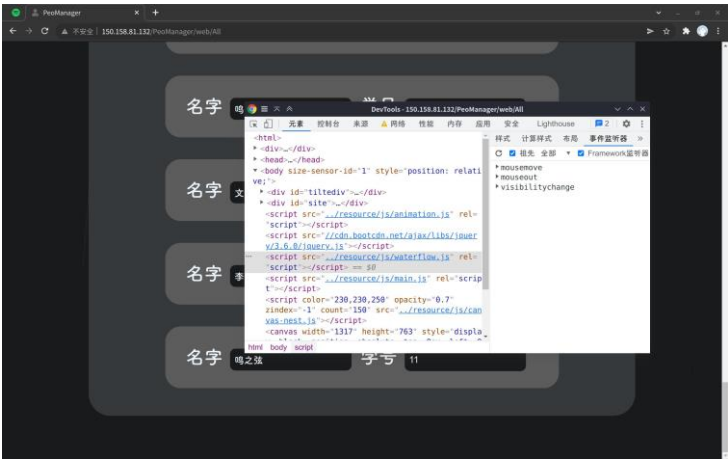


在查找结果与所有记录查看界面中使用了瀑布流效果，浏览器会监听鼠标滚动，判断是否接近底部，并适时出发异步加载数据，在所有结果加载完成后会关闭滚动监听：

时间监听器中存在滚动监听：



监听关闭后：



## 5.2 系统实现的不足

1. 系统在安全性方面仍存在不足之处。
2. 系统在前后端分离方面做的较差，因项目较小，且需求仅为 web 网站，故便携时前后端结合较为紧密，对其他平台兼容性较差，项目整体的可移植性较差。
3. 系统在运行效率方面仍有很大进步空间，用户体验较差，如项目目前为服务器端处理完成后返回整个网页，但有些网页因处理方法较为复杂，返回时间较长，应采用优先返回网页框架以及少量信息，随后利用异步加载填充处理时间较长的网页，用户在看到网页加载的动画时体验要优于用户看到浏览器加载网页时的空白界面。
4. 项目扩展性较差，对于需要多种信息记录，如需同时记录姓名、id、性别、家乡等的场景适应性差，不能由管理员动态修改，导致项目复用性、扩展性差。