

- [README](#)
 - 1. 项目介绍
 - 2. 技术栈
 - 3. 项目结构
 - 4. 组员分工
 - 5. 项目部署
 - 5.1 数据库部署
 - 5.2 后端 SpringBoot 项目部署
 - 5.3 前端 Vue 项目部署
 - 6. 一些测试用例

README

1. 项目介绍

项目名: AttendanceSysWeb (员工考勤管理系统-Web版)

这是一套基于 SpringBoot + Vue.js 的前后端分离的员工考勤管理系统。某公司在软件研发与日常管理方面有着很成熟的管理方法，却没有一套效率高且方便的考勤系统，因此本系统意在帮助公司解决 **员工基本信息管理**、**员工上下班打卡**、**员工请假申请及请假审核**、**报表获取** 等问题，为公司对员工的考勤管理带来便利。

本系统主要有以下几个模块：

- 考勤管理模块
 - 员工与部门经理进行打卡签到签退
 - 人事部门获取月度报表
- 请假管理模块
 - 员工与部门经理管理自己的请假申请
 - 部门经理与总经理查看人员的请假申请
- 请假审核模块
 - 部门经理、总经理、人事部门对人员的请假申请进行审核
- 信息管理模块
 - 人员查看自己的基本信息
 - 人事部门管理人员的基本信息

2. 技术栈

- 后端所用技术
 - Java11

- SQL
- Spring Boot
- Spring Security
- MyBatis
- 前端所用技术
 - HTML
 - CSS
 - JavaScript
 - Vue.js
 - Bootstrap
- 开发工具
 - Visual Studio Code: 前端代码编辑
 - IntelliJ IDEA: 后端代码编辑及调试
 - Chrome: 前端调试
 - Postman: 接口测试
 - MySQL: 关系型数据库
 - Astah: UML 建模环境
 - Git/Gitee: 代码版本控制

3. 项目结构

- AttendanceSysWeb
 - **back-end**: 后端 Java 代码
 - AttendanceSysWeb: 后端项目根目录
 - **doc**: 相关文档
 - **front-end**: 前端代码
 - AttendanceSysWeb: 前端项目根目录
 - **sql**: SQL 脚本
 - create_authen_table.sql: 建立鉴权表
 - create_base_table.sql: 建立基本表

后端项目结构:

- .idea
- .mvn
- src:
 - main:
 - java: 源码根目录
 - com
 - zmy
 - demo
 - bean: Java Bean 定义

- **controller**: 控制层
- **repository**: 持久化层
- **sans**: 鉴权
- **service**: 业务逻辑层
- **util**: 工具包
- **DemoApplication.java**: 启动类
- **Test.java**
- **resource**: 资源文件根目录
 - **credential**: 用户文件根目录
 - **mapper**: MyBatis 映射文件 (SQL 语句)
 - **application-dev.yml**: Spring Boot 配置文件
 - **application.yml**: Spring Boot 配置文件
 - **lombok.config**

4. 组员分工

组员	分工
黄泓 (组长)	<ul style="list-style-type: none"> • 环境配置 • 数据库搭建 • 后端项目框架搭建 • 核心部分代码编写 (<code>bean</code>、<code>controller</code>、<code>repository</code>、<code>service</code>、<code>util</code>) • SQL 语句编写 (<code>resource/mapper</code>)
叶紫钰	<ul style="list-style-type: none"> • 鉴权部分代码编写 (<code>sans</code>) • 接口测试 • 项目部署
周陈铮	<ul style="list-style-type: none"> • 环境配置 • 前端项目框架搭建 • 页面布局 • 核心组件
王迪	<ul style="list-style-type: none"> • 报表组件 • 签到签退组件 • 个人信息管理组件

5. 项目部署

- 本项目基于 Windows10 系统运行

5.1 数据库部署

1. 确保 MySQL 已安装 (v8.0.25) 并启动服务。
2. 在 MySQL 中建立数据库 `attendance_system`。
3. 使用 MySQL 运行 `./sql/` 目录下的 `create_base_table.sql` 与 `create_authen_table.sql` 脚本, 成功建立数据表。
4. 可使用 `show tables;` 命令查看 `attendance_system` 数据库下的数据表, 正确情况应该有 14 张表。

5.2 后端 SpringBoot 项目部署

1. 确保 Java 11 已安装并设置环境变量。
2. 使用 IDEA 打开后端项目根目录 `./back-end/AttendanceSysWeb/`, 设置项目 JDK。
3. 修改 `src/resource/application-dev.yml` 文件下的 `spring.datasource.username` 与 `spring.datasource.password` 字段为自己本地 MySQL 的用户名与密码。
4. 运行 `DemoApplication.java`, 项目运行在 8080 端口。

5.3 前端 Vue 项目部署

1. 确保已正确安装 Node.js (v12.16.1) 并配置好环境变量, 使用命令 `node -v` 查看是否正确安装。
2. 使用 `cmd` 打开前端项目根目录 `./front-end/AttendanceSysWeb/`。
3. 配置淘宝镜像

```
npm config set registry https://registry.npm.taobao.org
npm config get registry
```

第2条命令输出结果应为: `https://registry.npm.taobao.org/`

4. 配置 npm 下载依赖位置 (位置可自定义) :

```
npm config set cache "node-repos/cache"
npm config set prefix "node-repos/prefix"
```

5. 使用 `npm config ls` 验证 Node.js 环境配置。
6. 安装 axios

```
npm install axios --save-dev
```

可以看到项目多了一个 `node_modules` 文件夹。

7. 编译运行 Vue 项目

```
npm run serve
```

8. 控制台上会打印出访问项目的 IP 地址及端口号, 如:

```
App running at:
- Local:   http://localhost:8081/
- Network: http://localhost:8081/

Note that the development build is not optimized.
To create a production build, run npm run build.
```

在浏览器中访问 `http://localhost:8081/` 即可。

6. 一些测试用例

用户名	密码	姓名	部门	角色
u5001	p5001	黄泓	综合部门	总经理
u5002	p5002	周陈铮	综合部门	人事部
u5003	p5003	菜虚琨	综合部门	部门经理
u5004	p5004	王冰冰	综合部门	普通员工
u1001	p1001	胡英俊	技术部门	部门经理
u1002	p1002	傻妞	技术部门	普通员工
u1003	p1003	柯洁	技术部门	普通员工
u2001	p2001	李云龙	运营部门	部门经理
u2002	p2002	工藤新一	运营部门	普通员工
u3001	p3001	papi酱	产品部门	部门经理
u3002	p3002	川宝	产品部门	普通员工
u4001	p4001	常山赵子龙	市场部门	部门经理
u4002	p4002	奶茶妹妹	市场部门	普通员工

注：因为签到签退的地理范围中心设置的是 天津大学北洋园校区，故在其他地方运行本系统会无法正常签到签退。