# 引言

## 1.1标识

标题：《总体设计说明书》

简称：SDS

版本号：1.0.0

说明：本文档支持Office、WPS等word文档阅读软件

## 1.2系统概述

国内的中小微企业，在招聘环节，他们都面临着招工难、踩坑多的困扰。招到一个缺乏责任感、能力不足、稳定性差的员工，对企业和团队来说，都将造成巨大损失。因此，需要建立一套跨组织人才管理系统，可以帮助新雇主客观全面真实地了解面试者曾经的工作能力与工作状态，以便更好地匹配岗位，并规避招错人的风险。

## 1.3文档概述

本文档是对该开发系统的一个总体的把握，旨在推动该项目软件工程的规范化，使设计人员遵循统一的详细设计书写规范，节省制作文档的时间，降低系统实现的风险，做到系统设计资料的规范性与全面性，以利于系统的实现、测试、维护、版本升级等。

## 1.4参考资料

《Software Engineering》 出版社：机械工业出版社 作者：Roger S.Pressman

《SQL编程基础》 出版社: 机械工业出版社作者：Patrick.J.J.

《深入浅出——MySQL开发、管理与应用实例》出版社：清华大学出版社 作者：邹建

《数据库系统概论》 出版社：高等教育出版社 作者：王珊，萨师煊

# 总体设计

## 2.1需求规定

#### 功能：

**用户方：**

1. 求职
2. 离职
3. 签到
4. 查看任务
5. 查看个人简历
6. 查看个人信息
7. 修改个人信息
8. 登录

**HR方：**

1. 求职

2. 离职

3. 签到

4. 查看任务

5. 查看个人简历

6. 查看个人信息

7. 修改个人信息

8. 登录

9. 评价

10. 同意用户求职

11. 同意用户离职

**主管方：**

1. 发布任务
2. 评价
3. 同意离职
4. 任命HR
5. 登录
6. 修改公司信息
7. 修改个人信息

**管理员方：**

1. 新增公司
2. 新增公司主管
3. 修改公司信息
4. 修改主管信息

#### 性能：

该系统在日常处理中的响应速度为秒级，达到实时要求，以及时反馈信息。在进行统计分析时，根据所需数据量的不同而从秒级到分钟级，原则是保证操作人员不会因为速度问题而影响工作效率。

#### 数据来源：

用户手动输入的数据

#### 输入要求：

1. 注册账号时不能与其他用户重复
2. 用户登录账号时，账号需详实准确
3. 用户登录账号时，账号与密码应匹配
4. 用户搜索时输入的关键字不得包含敏感词汇
5. 用户编写简历或评价时不得包含敏感词汇

#### 输出要求：

1. 显示的信息中不能包含错别字
2. 显示的信息中不能包含敏感词汇

## 2.2运行环境

设备数量：三台小组成员pc

处理器：无特殊需求，现在市面上的硬件设备性能完全可以支撑使用。

存储器：8G及以上运行内存

输入/输出设备：鼠标、键盘、显示器、USB接口

辅助存储器：128G以上内存

通信/网络设备：网络驱动器，能正常连网即可

## 2.3供选择的方案

**2.3.1 供选择的方案1**

方案：

采用jsp框架+ tomcat服务器部署，数据库采用mysql

缺点：

1. jsp耦合度太高、前后端不分离，属于过时的技术。
2. 如今大部分项目同时使用多种数据库，完成访问速度。
3. Jsp需要部署tomcat服务器。

优点：

不需要前后端沟通

存在问题：

环境搭建较难。

经济可行性：

支出：

1. 云服务器租金
2. Office正版费用
3. Windows 10 正版费用

收入：

1. 广告收入
2. 按照网站人数增加带来的一系列收益，包括但不限于：会员收益等。

**2.3.2 供选择的方案2**

方案：

用html+css制作前端，用java web实现后端，用MySQL做数据库。

优点：

基本能实现预期效果。

能将小组的学习成本降到最低

缺点：

界面美观程度欠缺，用户体验可能不够好

局限性及存在的问题：

用户体验问题

经济可行性：

支出：

1. 云服务器租金
2. Office 正版费用
3. Windows 10 正版费用

收入：

1. 广告收入
2. 按照网站人数增加带来的一系列收益，包括但不限于：会员收益等。

**2.3.3 供选择的方案3**

方案：

要用Vue制作前端，用Spring boot框架实现后端，用MySQL、Redis做数据库。

优点：

基本能实现预期效果。

Spring boot可以简化搭建环境，不需要部署tomcat，直接运行jar包

Redis数据库访问速度快

缺点：

组员所需自学的科目过多，时间上可能来不及；

局限性及存在的问题：

大部分内容涉及到的技术都没有学过，需要自学；

经济可行性：

支出：

1. 云服务器租金
2. Office 正版费用
3. Windows 10正版费用

收入：

1. 广告收入
2. 按照网站人数增加带来的一系列收益，包括但不限于：会员收益等。

**最终经过小组商讨与权衡，决定采用方案3。**原因如下：

方案1的技术有些过时，前后端不分离不需要沟通，但是耦合度太高，环境的搭建也较为困难，可能会影响到后续的实现和测试

方案2技术难度相对较低，但是存在用户体验较差的问题。

方案3前后端分离，耦合度较低，采用框架也能优化用户体验。

## 2.4基本设计概念和处理流程

### 2.4.1系统流程图

### 2.4.2数据流图

# 3.接口设计

## 3.1用户接口

1.将要采用的图形用户界面（GUI）标准。

2.屏幕布局自适应。

3.每个屏幕的标准按钮（参考UI界面图）；

4.错误信息显示标准——提示框跳出错误信息。

## 3.2外部接口

1. 将要采用的图形用户界面（GUI）标准

2. 屏幕布局自适应

3. 每个屏幕的标准按钮（参考UI界面图）

4. 错误信息显示标准——提示框跳出错误信息

## 3.3内部接口

TBD

# 4.运行设计

## 4.1运行模块组合

### 4.1.1用户功能模块组合

### 4.1.2HR功能模块组合

### 4.1.3主管功能模块组合

### 4.1.4管理员功能模块组合

## 4.2运行控制