****

**“兼职桥”**

**项**

**目**

**需**

**求**

**分**

**析**

**报**

**告**

目录

[**1. 引言** 3](#_Toc181110384)

[**1.1 项目背景** 3](#_Toc181110385)

[**1.2 目的** 3](#_Toc181110386)

[**1.3 范围** 3](#_Toc181110387)

[**2. 总体描述** 3](#_Toc181110388)

[**2.1 产品概述** 3](#_Toc181110389)

[**2.2 系统环境** 3](#_Toc181110390)

[**2.3 用户特征** 4](#_Toc181110391)

[**2.4 假设和约束** 4](#_Toc181110392)

[**3. 功能需求** 4](#_Toc181110393)

[**3.1 用户注册和登录系统** 4](#_Toc181110394)

[**3.2** **雇主发布兼职信息** 4](#_Toc181110395)

[**3.3 求职者浏览和申请兼职工作** 4](#_Toc181110396)

[**3.4 消息通知系统** 4](#_Toc181110397)

[**3.5 严格控制用户权限** 4](#_Toc181110398)

[**3.6 提供个性化匹配** 5](#_Toc181110399)

[**3.7 供需双方互相打分评价** 5](#_Toc181110400)

[**3.8 个人空间** 5](#_Toc181110401)

[**4. 非功能需求** 5](#_Toc181110402)

[**4.1 响应速度** 5](#_Toc181110403)

[**4.2 安全需求** 5](#_Toc181110404)

[**4.3 可用性需求** 6](#_Toc181110405)

[**4.4 可扩展性需求** 6](#_Toc181110406)

[**4.5 兼容性需求** 6](#_Toc181110407)

[**5. 用户需求** 6](#_Toc181110408)

[**5.1 用户目标** 6](#_Toc181110409)

[**5.2 用户场景** 7](#_Toc181110410)

[**6. 数据需求** 7](#_Toc181110411)

[**6.1 用户界面设计** 7](#_Toc181110412)

[**6.2 前端功能需求** 7](#_Toc181110413)

[**6.3 前端技术需求** 8](#_Toc181110414)

[**7. 数据库设计需求** 8](#_Toc181110415)

[**7.1 数据库模型** 8](#_Toc181110416)

[**7.2 数据库功能** 8](#_Toc181110417)

[**7.3 数据库技术** 8](#_Toc181110418)

[**8. 安全性和维护需求** 9](#_Toc181110419)

[**8.1 安全性需求** 9](#_Toc181110420)

[**8.2 维护需求** 9](#_Toc181110421)

[**9. 附录** 9](#_Toc181110422)

[**9.1 参考文献** 9](#_Toc181110423)

**1. 引言**

**1.1 项目背景**

随着互联网的普及和远程工作的兴起，越来越多的人开始寻找兼职工作机会。兼职桥旨在提供一个平台，让雇主能够发布兼职信息，同时让求职者能够轻松找到合适的兼职机会。

**1.2 目的**

业务目标：

成为领先的在线兼职信息发布平台。

在第一年内吸引至少1万名注册用户。

实现至少500个成功匹配的兼职工作。

技术目标：

开发一个用户友好、响应迅速的平台。

实现高可用性和安全性。

提供个性化推荐系统。

**1.3 范围**

包括平台的开发、测试、上线和维护。 不包括实体办公场所的建立和传统媒体广告。

**2. 总体描述**

**2.1 产品概述**

概述系统或软件的主要功能、目标用户及核心价值。说明系统解决的问题及其预期效果。

**2.2 系统环境**

说明系统的运行环境，包括硬件、软件、网络、开发环境等。

**2.3 用户特征**

描述目标用户的特征，包括用户类型（普通用户、管理员等）、用户数量、用户技术水平等。

**2.4 假设和约束**

列出在需求分析中涉及的假设条件和项目的约束条件。

**3. 功能需求**

**3.1 用户注册和登录系统**

允许用户创建个人账户并安全登录，以访问网站的全部功能。系统将验证用户提供的信息，并确保账户的安全性。

* 1. **雇主发布兼职信息**

雇主可以创建并发布兼职职位信息，包括职位描述、要求、薪资等。系统将提供界面以便于信息的输入和管理。

**3.3 求职者浏览和申请兼职工作**

求职者可以浏览可用的兼职职位，并根据自己的条件和偏好申请合适的工作。系统将提供搜索和过滤功能以便于职位查找。

**3.4 消息通知系统**

系统将通过电子邮件或网站内通知提醒用户有关新发布的工作、申请状态更新或其他重要信息。

**3.5 严格控制用户权限**

系统将根据用户的角色（求职者或雇主）实施不同的访问控制，确保每个用户只能访问和操作其权限范围内的功能。

**3.6 提供个性化匹配**

利用算法分析用户的偏好和历史行为，系统将推荐个性化的兼职职位给求职者，以及合适的候选人给雇主。

**3.7 供需双方互相打分评价**

完成兼职工作后，雇主和求职者可以互相评价对方的性能和合作体验。这有助于建立信任并提高平台的整体质量。

**3.8 个人空间**

用户可以创建个人资料，展示自己的技能、经验和成就。这为求职者提供了自我推广的机会，同时也让雇主能够更好地了解潜在雇员。

**4. 非功能需求**

**4.1 响应速度**

系统应能够快速响应用户的操作和请求，提供流畅的用户体验。这包括快速加载页面、即时搜索结果和高效的数据处理。

**4.2 安全需求**

确保用户数据的保护和隐私，通过加密技术、安全协议和访问控制来防止未授权访问和数据泄露。

**4.3 可用性需求**

系统应提供直观的用户界面和清晰的导航，以帮助用户轻松找到所需信息和完成操作。此外，系统应提供帮助文档和客户支持。

**4.4 可扩展性需求**

系统设计应允许轻松添加新功能和处理增加的用户负载，以适应未来业务增长和技术发展。

**4.5 兼容性需求**

网站应兼容主流的浏览器和设备，包括桌面和移动设备，确保所有用户都能无缝访问和使用网站的功能。

**5. 用户需求**

**5.1 用户目标**

大学生兼职发布平台的用户目标主要分为两类：求职者和雇主。

* **求职者目标**：
  + 寻找合适的兼职工作以获得额外收入或工作经验。
  + 通过平台的个性化匹配功能快速找到符合自己技能和兴趣的职位。
  + 轻松申请职位并通过消息通知系统及时了解申请状态。
  + 通过评价系统了解雇主的信誉和其他求职者的经验。
* **雇主目标**：
  + 发布兼职职位信息以吸引合适的候选人。
  + 利用平台的个性化匹配功能找到符合职位要求的求职者。
  + 通过消息通知系统与求职者进行沟通。
  + 通过评价系统建立良好的雇主声誉，吸引更多优秀求职者。

**5.2 用户场景**

在校大学生浏览网站寻找兼职工作、企业通过平台发布工作、供需双方互相评价

**6. 数据需求**

**6.1 用户界面设计**

* **响应式设计**：网站应兼容各种设备，包括桌面、平板和手机。
* **访问速度**：优化页面加载速度，确保快速响应。
* **导航结构**：清晰的导航菜单，方便用户快速找到所需信息。
* **用户注册/登录界面**：简洁明了的注册和登录流程，支持社交媒体账号快速登录。
* **职位发布界面**：提供直观的表单，方便雇主发布和管理兼职职位。
* **简历上传界面**：允许求职者上传简历，提供简历填写向导。
* **搜索和筛选功能**：提供高级搜索选项，允许用户根据关键词、类别、地点和薪资筛选职位。
* **个人空间**：用户可以自定义个人空间，展示个人或公司信息。
* **消息通知**：实时通知系统，提醒用户新的职位申请、雇主回复等。

**6.2 前端功能需求**

* **表单验证**：在客户端进行输入验证，确保数据的正确性和完整性。
* **实时搜索建议**：在用户输入搜索关键词时提供实时建议。
* **地图集成**：在职位列表中集成地图视图，显示工作地点。
* **多媒体支持**：允许在职位描述和公司介绍中嵌入图片和视频。
* **聊天功能**：内置聊天系统，支持雇主和求职者之间的即时通讯。
* **多语言支持**：网站应支持多种语言，以满足不同用户的需求。
* **无障碍设计**：确保网站符合无障碍标准，方便残障人士使用。

**6.3 前端技术需求**

* **前端框架**：使用Vue.js或React等现代JavaScript框架提高开发效率和性能。
* **状态管理**：使用Redux或Vuex等状态管理库，以管理应用状态。
* **SEO优化**：遵循SEO最佳实践，提高网站在搜索引擎中的排名。
* **代码规范**：遵循一致的编码标准和命名约定，确保代码的可读性和可维护性。
* **性能监控**：集成前端性能监控工具，实时跟踪网站性能。

**7. 数据库设计需求**

**7.1 数据库模型**

* **数据一致性**：确保数据库设计满足第三范式，避免数据冗余。
* **数据完整性**：实施数据完整性约束，如外键、唯一性和非空约束。
* **索引优化**：为频繁查询的列创建索引，提高查询效率。
* **备份和恢复**：定期备份数据库，并确保可以快速恢复数据。

**7.2 数据库功能**

* **用户管理**：存储用户信息，支持用户注册、登录和个人信息更新。
* **职位管理**：存储和管理兼职职位信息，包括职位描述、薪资和地点等。
* **简历管理**：存储求职者的简历信息，支持简历的上传和下载。
* **申请跟踪**：记录职位申请信息，包括申请状态和申请时间。
* **消息系统**：存储用户之间的消息往来，支持消息的发送和接收。
* **通知系统**：存储系统通知信息，支持通知的发送和接收。

**7.3 数据库技术**

* **数据库系统选择**：选择MySQL或PostgreSQL等关系数据库管理系统。
* **查询优化**：编写高效的SQL查询，减少数据库的负载。
* **数据安全**：实施数据加密和安全措施，保护敏感数据不被泄露。
* **连接池**：使用数据库连接池技术，提高数据库连接的效率和性能。
* **分页和排序**：实现数据的分页和排序功能，提高大数据量处理的性能。

**8. 安全性和维护需求**

**8.1 安全性需求**

* **数据加密**：对敏感数据进行加密存储，使用SSL/TLS保护数据传输。
* **用户认证和授权**：实现角色基础的访问控制，确保用户只能访问授权资源。
* **输入验证**：防止SQL注入和跨站脚本攻击（XSS），通过前端和后端的输入验证。

**8.2 维护需求**

* **日志记录**：记录系统操作日志，用于问题诊断和性能监控。
* **性能监控**：定期检查数据库性能，优化慢查询。
* **软件更新**：定期更新前端和后端软件，修补安全漏洞。

**9. 附录**

**9.1 参考文献**

* 杭州IT行业年平均工资hznews.hangzhou.com.cn/jingji/content/2023-05/09/content\_8530093.htm
* [揭秘软件开发的基石：瀑布模型全解析 (baidu.com)](https://cloud.baidu.com/article/3333602)
* [REST API设计分析及实证研究 (jos.org.cn)](https://jos.org.cn/jos/article/abstract/6383?st=online_first)
* [Fielding Dissertation: CHAPTER 5: Representational State Transfer (REST) (uci.edu)](https://ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm)
* 需求分析报告模板：GB-856T-88