九江学院

计算机与大数据科学学院

**《软件工程课程设计》报告书**

**题 目： 基于python的智能化招聘数据爬取与分析系统**

**专 业： 软件工程**

**班 级： 计科A2351**

**姓 名： 傅森林**

**学 号： 23**

**指导老师： 王超 王江涛**

**设计时间： 2025年3月3日 ~ 2025年3月7日**

**目 录**

[《软件工程课程设计》任务书 1](#_Toc9674)

[1. 课程设计（实训）计划 3](#_Toc11418)

[2. 基于Git的开发流程 3](#_Toc24656)

[3. 描述编程技术在软件开发实践中的综合运用 4](#_Toc21234)

[4. Python开发技术的拓展 5](#_Toc24365)

[5. 实践心得体会与总结 6](#_Toc9670)

[本科课程实训量化评分标准 7](#_Toc2319)

九江学院

计算机与大数据科学学院

## 《软件工程课程设计》任务书

**设计题目**： 基于python的智能化招聘数据爬取与分析系统

**专 业：** 软件工程

**班 级：** 计科A2351/52/53

**姓 名:**  傅森林 **学 号:** 23

**起止日期:** 2025年3月3日~2025年3月7日

**指导教师:** 王超 王江涛

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 课程设计题目 | | | | 基于python的智能化招聘数据爬取与分析系统 | | | | | |
| 姓名 | 傅森林 | | | | 学号 | 23 | | 班级 | 计科A2351/52/53 |
| 组别 | 2 | | | | 专业 | 软件工程 | | | |
| 组员 | 张海源，樊登，罗春林 | | | | | | | | |
| 指导教师 | | | | 王超 王江涛 | | | | | |
| 课程设计目的 | 增强学生的软件开发的规范化意识，养成良好的编程习惯。使学生学习掌握GIT 项目管理工具。使学生掌握python爬虫技术的运用；对企业主流软件开发技术进行项目实训，增强学生对专业技能知识体系的应用能力；培养学生团队协作沟通的意识与能力。 | | | | | | | | |
| 课程设计环境 | | 硬件：计算机  软件：GIT 项目管理、Python中的面向对象、python正则表达式运用、Python中的request库与beautifulsoup库、MySQL数据库开发技术、Python中的GUI库、echars图库运用、Jmeter测试工具 | | | | | | | |
| 课程设计任务和要求 | | 1. 完成以下内容：项目开发流程与规范、项目开发的实践、软件测试。   2、 完成本课程实训报告书。 | | | | | | | |
| 课程设计工作进度计划 | | | | | | | | | |
| 序号 | | | 起止日期 | | | | 工 作 内 容 | | |
| 1 | | | 2025.3.3 | | | | 软件开发流程与规范 | | |
| 2 | | | 2025.3.4~2025.3.6 | | | | 软件开发实践 | | |
| 3 | | | 2025.3.7 | | | | 实践项目分析与总结 | | |
|  | | |  | | | |  | | |
|  | | |  | | | |  | | |
|  | | |  | | | |  | | |
|  | | |  | | | |  | | |
| 教研室审核意见：  教研室主任签字： 年 月 日 | | | | | | | | | |
| 院（系）审核意见：  签章： 年 月 日 | | | | | | | | | |

|  |
| --- |
| 1. 课程设计（实训）计划 2025.3.3 软件项目开发流程与GIT 项目管理规范；  2025.3.4 完成项目中对在线招聘网数据的爬取与分析功能；  2025.3.5 创建项目数据库，新建数据表，完成项目中爬取数据并保存至数据库的功能与数据显示功能；  2025.3.6 完成在线招聘网数据爬取及图表显示功能；使用jmeter进行增加数据、显示数据等接口测试；  2025.3.7 实践项目分析与总结。 |
| 2. 基于Git的开发流程 (主要内容：阐述相关项目的软件开发流程、包括需求分析、总体设计等内容)  项目流程：  基于python的智能化招聘数据爬取与分析系统》 是基于python语言对在线招聘网爬取数据,再对数据进行深度优先算法分析、数据清洗、数据展示的一个项目。其中爬取的信息有企业岗位信息、薪水信息、城市分布信息等，用到了python中的request库以及beautifulsoup库，并使用了正则表达式，以及当前主流的python爬取库和数据库开发技术，最后通过echars图表技术实现数据可视化。  对于功能测试，使用自动化测试工具jmeter对项目进行接口测试，压力测试。  具体内容：  1.需求描述：进行用例图的需求描述；完成1-2个用例的用例规约描述；  2.总体设计：进行顺序图的项目功能活动的描述；完成数据爬取的活动流程的顺序图；  **软件开发流程阐述**  **一、需求分析**   1. **需求收集与理解** 项目旨在开发基于 Python 的智能化招聘数据爬取与分析系统，核心需求是从在线招聘网获取企业岗位信息、薪水信息、城市分布信息等，并进行后续处理与展示。通过与相关利益者（如潜在用户、项目发起人）沟通，明确系统需具备数据爬取、分析、清洗、展示以及接口支持等功能。 2. **需求描述**    * **用例图描述**：绘制用例图，展示系统的参与者（如爬虫程序、数据分析模块、数据展示界面等）与系统功能用例之间的关系。例如，爬虫程序作为参与者，与数据爬取用例相关联；数据分析模块与深度优先算法分析用例相关联；数据展示界面与数据展示用例相关联。    * **用例规约描述**：选取 1 - 2 个关键用例，如“数据爬取用例”和“数据展示用例”进行详细规约描述。对于“数据爬取用例”，描述其前置条件（如网络连接正常、目标招聘网站可访问）、基本事件流（如使用 request 库发送请求获取网页内容，利用 BeautifulSoup 库和正则表达式解析数据）、后置条件（数据存储到数据库）；对于“数据展示用例”，描述其前置条件（数据库中有可展示的数据）、基本事件流（从数据库读取数据，使用 echarts 技术生成图表展示）、后置条件（用户看到可视化数据）。   **二、总体设计**   1. **功能活动描述** 采用顺序图描述项目功能活动，清晰展示各个功能模块之间的交互顺序。从数据爬取开始，爬虫程序使用 request 库向在线招聘网站发送请求，获取网页 HTML 内容，然后利用 BeautifulSoup 库和正则表达式解析出企业岗位信息、薪水信息、城市分布信息等，将解析后的数据存储到数据库。接着，数据分析模块从数据库读取数据进行深度优先算法分析，处理后的数据再次存储到数据库。最后，数据展示模块从数据库获取数据，使用 echarts 技术生成可视化图表展示给用户。 2. **数据爬取活动流程顺序图** 绘制数据爬取的活动流程顺序图，详细展示爬虫程序与目标招聘网站、数据库之间的交互过程。首先，爬虫程序初始化，设置目标网站 URL 和请求头等信息。然后，发送请求获取网页内容，若请求成功，解析网页内容提取数据；若请求失败，进行错误处理（如重试或记录错误日志）。解析出的数据经过格式化处理后，存储到数据库，完成数据爬取活动流程。 |
| 3. 描述编程技术在软件开发实践中的综合运用 (主要内容：阐述Python中的request库和beautiful库在实际开发中的运用；阐述数据库的设计和实现，如项目使用企业级MySQL数据库环境搭建与配置；创建项目数据库，新建招聘信息的数据表；对招聘信息表使用进行SQL语句操作；对爬取的数据存放进项目招聘信息表)  具体要求：   1. 详细设计：数据库设计与实现如：概念模型ER图、逻辑表结构，SQL语句创建表等 2. 核心功能的编码实现：阐述Python中的request库和beautiful库在实际开发中的运用的流程图以及核心代码的展示   **（一）概念模型 ER 图**  在招聘数据爬取与分析系统中，概念模型主要涉及实体及其关系。主要实体有企业、岗位。企业与岗位是一对多的关系，即一个企业可以发布多个岗位。   * **企业实体**：包含企业 ID（主键）、企业名称、企业简介等属性。 * **岗位实体**：包含岗位 ID（主键）、岗位名称、薪水、城市、企业 ID（外键）等属性。   **（二）逻辑表结构**  基于 ER 图，设计逻辑表结构如下：   1. **企业表（companies）** | 字段名称 | 数据类型 | 描述 | | - | - | - | | company\_id | INT | 企业 ID，主键，自增 | | company\_name | VARCHAR(255) | 企业名称 | | company\_intro | TEXT | 企业简介 | 2. **岗位表（jobs）** | 字段名称 | 数据类型 | 描述 | | - | - | - | | job\_id | INT | 岗位 ID，主键，自增 | | job\_name | VARCHAR(255) | 岗位名称 | | salary | VARCHAR(100) | 薪水 | | city | VARCHAR(100) | 城市 | | company\_id | INT | 企业 ID，外键，关联 companies 表的 company\_id |   **（三）SQL 语句创建表**  sql   |  |  | | --- | --- | |  | -- 创建企业表 | |  | CREATE TABLE companies ( | |  | company\_id INT AUTO\_INCREMENT PRIMARY KEY, | |  | company\_name VARCHAR(255) NOT NULL, | |  | company\_intro TEXT | |  | ); | |  |  | |  | -- 创建岗位表 | |  | CREATE TABLE jobs ( | |  | job\_id INT AUTO\_INCREMENT PRIMARY KEY, | |  | job\_name VARCHAR(255) NOT NULL, | |  | salary VARCHAR(100), | |  | city VARCHAR(100), | |  | company\_id INT, | |  | FOREIGN KEY (company\_id) REFERENCES companies(company\_id) | |  | ); |   **二、核心功能的编码实现**  **（一）request 库和 beautiful 库运用流程图**  mermaid   |  |  | | --- | --- | |  | flowchart TD | |  | A[开始] --> B[导入 request 和 BeautifulSoup 库] | |  | B --> C[设置目标 URL 和请求头] | |  | C --> D[发送请求获取网页内容] | |  | D --> E{请求是否成功?} | |  | E -- 是 --> F[解析网页内容] | |  | E -- 否 --> G[记录错误日志并结束] | |  | F --> H[提取招聘信息] | |  | H --> I[将信息存储到数据库] | |  | I --> J[结束] |   **（二）核心代码展示**  python   |  |  | | --- | --- | |  | import requests | |  | from bs4 import BeautifulSoup | |  | import pymysql | |  |  | |  | # 数据库连接信息 | |  | db\_config = { | |  | 'host': 'localhost', | |  | 'user': 'your\_username', | |  | 'password': 'your\_password', | |  | 'database': 'your\_database' | |  | } | |  |  | |  | # 目标 URL 和请求头 | |  | url = 'https://www.example.com/jobs' | |  | headers = { | |  | 'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.3'} | |  |  | |  | try: | |  | # 发送请求 | |  | response = requests.get(url, headers=headers) | |  | response.raise\_for\_status() # 如果请求失败，抛出异常 | |  |  | |  | # 解析网页内容 | |  | soup = BeautifulSoup(response.text, 'html.parser') | |  |  | |  | # 提取招聘信息（这里以示例为准，实际需根据网页结构调整） | |  | job\_list = soup.find\_all('div', class\_='job-item') | |  | for job in job\_list: | |  | job\_name = job.find('h2').text.strip() | |  | salary = job.find('span', class\_='salary').text.strip() | |  | city = job.find('span', class\_='city').text.strip() | |  | # 这里假设企业信息在另一个标签中，简单示例 | |  | company\_name = job.find('div', class\_='company-name').text.strip() | |  |  | |  | # 连接数据库 | |  | connection = pymysql.connect(\*\*db\_config) | |  | try: | |  | with connection.cursor() as cursor: | |  | # 插入企业信息（简单处理，实际可能需先查询是否存在） | |  | cursor.execute("INSERT INTO companies (company\_name) VALUES (%s)", (company\_name,)) | |  | company\_id = cursor.lastrowid | |  |  | |  | # 插入岗位信息 | |  | cursor.execute("INSERT INTO jobs (job\_name, salary, city, company\_id) VALUES (%s, %s, %s, %s)", | |  | (job\_name, salary, city, company\_id)) | |  | connection.commit() | |  | finally: | |  | connection.close() | |  |  | |  | except requests.RequestException as e: | |  | print(f"请求失败: {e}") | |  | except Exception as e: | |  | print(f"发生错误: {e}") |   在上述代码中，首先导入所需的库，然后设置目标 URL 和请求头。通过 requests.get 发送请求获取网页内容，使用 BeautifulSoup 解析网页。接着从解析后的内容中提取招聘信息，如岗位名称、薪水、城市和企业名称。最后，连接 MySQL 数据库，将提取的信息分别插入到企业表和岗位表中。  通过以上数据库设计和核心功能编码实现，展示了 Python 中的 request 库和 beautiful 库在实际开发中的综合运用，以及企业级 MySQL 数据库的设计和使用。 |
| 4. Python开发技术的拓展 (主要内容：依据所了解的python的开发技术栈，阐述相关技术以及应用范围，Python读取mysql数据库中数据；Python中GUI库的使用；echars图库的运用jmeter测试工具的使用。)  编码实现：数据库中的数据读取、数据界面展示；  测试：运用测试工具，完成的项目的功能测试；  **一、相关技术及应用范围**  **（一）Python 读取 MySQL 数据库中数据**  Python 可以通过多种库来连接和读取 MySQL 数据库，常用的库有 pymysql 和 mysql - connector - python。其应用范围广泛，例如在 Web 开发中从数据库获取用户信息、订单信息等；在数据分析领域，从数据库读取数据进行清洗、分析和可视化；在自动化运维中，获取系统配置信息等。  **（二）Python 中 GUI 库的使用**  Python 提供了多个 GUI 库，如 tkinter、PyQt、wxPython 等。tkinter 是 Python 标准库中自带的 GUI 库，简单易用，适合创建小型的桌面应用程序，如简单的文本编辑器、计算器等。PyQt 功能强大，提供了丰富的控件和工具，可用于开发复杂的跨平台桌面应用，如音乐播放器、图像编辑软件等。wxPython 也具有良好的跨平台特性，其界面风格更接近原生系统，常用于开发企业级的桌面应用。  **（三）echarts 图库的运用**  Echarts 是一个基于 JavaScript 的开源可视化图表库，Python 可以通过 pyecharts 库来使用其功能。应用范围包括数据可视化展示，如展示销售数据的趋势、不同地区的数据分布、用户行为分析等。可以生成柱状图、折线图、饼图、地图等多种类型的图表，直观地呈现数据特征和规律。  **（四）jmeter 测试工具的使用**  Jmeter 是一款开源的性能测试工具，主要用于对软件进行压力测试、负载测试、接口测试等。在 Web 应用开发中，可以模拟大量用户并发访问，测试服务器的性能和稳定性；在接口开发中，验证接口的正确性，检查接口的响应时间、吞吐量等性能指标。  **二、编码实现**  **（一）数据库中的数据读取**  python   |  |  | | --- | --- | |  | import pymysql | |  |  | |  | # 数据库连接信息 | |  | db\_config = { | |  | 'host': 'localhost', | |  | 'user': 'your\_username', | |  | 'password': 'your\_password', | |  | 'database': 'your\_database' | |  | } | |  |  | |  | try: | |  | # 连接数据库 | |  | connection = pymysql.connect(\*\*db\_config) | |  | with connection.cursor() as cursor: | |  | # 执行 SQL 查询 | |  | sql = "SELECT job\_name, salary, city FROM jobs" | |  | cursor.execute(sql) | |  | results = cursor.fetchall() | |  | for row in results: | |  | print(f"岗位名称: {row[0]}, 薪水: {row[1]}, 城市: {row[2]}") | |  | except pymysql.MySQLError as e: | |  | print(f"数据库操作错误: {e}") | |  | finally: | |  | if connection: | |  | connection.close() |   **（二）数据界面展示（使用 tkinter）**  python   |  |  | | --- | --- | |  | import pymysql | |  | import tkinter as tk | |  | from tkinter import ttk | |  |  | |  | # 数据库连接信息 | |  | db\_config = { | |  | 'host': 'localhost', | |  | 'user': 'your\_username', | |  | 'password': 'your\_password', | |  | 'database': 'your\_database' | |  | } | |  |  | |  | class JobDisplayApp: | |  | def \_\_init\_\_(self, root): | |  | self.root = root | |  | self.root.title("招聘信息展示") | |  |  | |  | self.tree = ttk.Treeview(root, columns=('job\_name', 'salary', 'city'), show='headings') | |  | self.tree.heading('job\_name', text='岗位名称') | |  | self.tree.heading('salary', text='薪水') | |  | self.tree.heading('city', text='城市') | |  | self.tree.pack() | |  |  | |  | self.fetch\_data() | |  |  | |  | def fetch\_data(self): | |  | try: | |  | connection = pymysql.connect(\*\*db\_config) | |  | with connection.cursor() as cursor: | |  | sql = "SELECT job\_name, salary, city FROM jobs" | |  | cursor.execute(sql) | |  | results = cursor.fetchall() | |  | for row in results: | |  | self.tree.insert('', 'end', values=row) | |  | except pymysql.MySQLError as e: | |  | print(f"数据库操作错误: {e}") | |  | finally: | |  | if connection: | |  | connection.close() | |  |  | |  | if \_\_name\_\_ == "\_\_main\_\_": | |  | root = tk.Tk() | |  | app = JobDisplayApp(root) | |  | root.mainloop() |   **三、测试**  **（一）接口测试**  使用 jmeter 对项目的接口进行测试，例如对数据获取接口进行测试。   1. **创建测试计划**：打开 jmeter，新建一个测试计划。 2. **添加线程组**：在测试计划下添加一个线程组，设置线程数、循环次数等参数，模拟不同数量的用户请求。 3. **添加 HTTP 请求**：在线程组下添加 HTTP 请求，设置请求的 URL、方法（如 GET）、请求头等信息。 4. **添加监听器**：添加查看结果树、聚合报告等监听器，用于查看测试结果，包括响应时间、响应状态码、吞吐量等。   **（二）功能测试**  除了接口测试，还可以使用 jmeter 进行功能测试，例如验证数据展示功能是否正常。可以通过模拟不同的输入条件，检查系统的输出是否符合预期。同时，结合日志记录和断言功能，进一步验证功能的正确性。  通过以上编码实现和测试过程，展示了 Python 开发技术栈中相关技术的应用和实践。 |
| 5. 实践心得体会与总结 (主要内容：课程设计实践过程中的体会以及总结)  在参与基于 Python 的智能化招聘数据爬取与分析系统课程设计实践的过程中，我收获颇丰，不仅加深了对 Python 开发技术栈的理解和运用，还积累了宝贵的项目实践经验。以下是我对此次实践的体会和总结。  **一、技术层面**  **（一）知识融合与运用**  此次实践让我深刻体会到 Python 开发技术栈中各项技术的融合与协作。从使用 request 库和 BeautifulSoup 库进行数据爬取，到运用 pymysql 库操作 MySQL 数据库存储和读取数据，再到利用 tkinter 实现数据界面展示、pyecharts 进行数据可视化，以及使用 jmeter 进行测试，每一项技术都在项目中发挥着不可或缺的作用。例如，在数据爬取阶段，需要合理设置请求头、处理请求异常，同时运用解析库准确提取所需信息；在数据库操作中，要设计合理的表结构，确保数据的完整性和一致性；在界面展示和可视化环节，要根据数据特点选择合适的图表类型和展示方式。这让我认识到，只有将各项技术熟练掌握并灵活运用，才能高效地完成一个完整的项目。  **（二）技术挑战与解决**  在实践过程中，我遇到了许多技术挑战。在数据爬取时，部分招聘网站采用了反爬机制，导致请求被拒绝。通过查阅资料和尝试不同的方法，我学会了设置代理 IP、模拟用户行为（如随机间隔请求时间）等方式来绕过反爬限制。在数据库设计方面，最初没有充分考虑数据的扩展性，导致后续添加新字段时出现困难。经过重新评估和设计，我学会了如何构建更合理的表结构，添加适当的索引以提高查询效率。这些挑战让我明白，技术的学习是一个不断探索和解决问题的过程，遇到问题时积极寻求解决方案，能够不断提升自己的技术能力。  **二、项目流程层面**  **（一）规范流程的重要性**  严格按照软件开发的流程进行实践，让我深刻认识到规范流程的重要性。从需求分析阶段明确项目的目标和功能，到总体设计阶段规划系统的架构和模块，再到详细设计、编码实现、测试和维护，每一个环节都紧密相连。在需求分析阶段，如果没有充分理解用户需求，可能会导致后续设计和编码出现偏差；在测试阶段，如果没有全面的测试计划，可能会遗漏一些潜在的错误。规范的流程能够帮助我们更好地组织项目，提高开发效率，保证项目质量。  **（二）团队协作与沟通**  虽然此次实践主要以个人形式进行，但在一些环节中也让我意识到团队协作和沟通的重要性。例如，在参考他人的代码和解决方案时，需要与他人进行有效的沟通和交流，理解他们的思路和方法。如果是在团队项目中，良好的团队协作和沟通能够避免重复工作，提高工作效率，及时解决项目中遇到的问题。因此，在今后的学习和工作中，我将更加注重培养自己的团队协作和沟通能力。  **三、个人成长层面**  **（一）自主学习能力的提升**  在实践过程中，遇到了许多之前未曾接触过的知识和问题，这促使我不断自主学习。通过查阅官方文档、技术论坛、博客等资源，我逐渐掌握了解决问题的方法。例如，在学习 pyecharts 库进行数据可视化时，通过阅读官方文档和参考他人的示例代码，我学会了如何创建各种类型的图表，并根据需求进行定制。这种自主学习能力的提升将对我今后的学习和工作产生积极的影响。  **（二）耐心和细心的培养**  项目开发是一个细致而复杂的过程，需要具备足够的耐心和细心。在编码过程中，一个小小的语法错误或逻辑错误都可能导致程序无法正常运行。通过不断地调试和排查问题，我逐渐培养了自己的耐心和细心，学会了认真对待每一个细节，确保代码的质量和稳定性。  **四、总结与展望**  通过此次课程设计实践，我不仅掌握了 Python 开发技术栈中的多项技术，还熟悉了软件开发的流程和方法，提升了自己的自主学习能力、解决问题的能力以及团队协作和沟通能力。同时，我也认识到自己存在的不足之处，例如在某些技术细节上还不够深入，项目经验还不够丰富。在今后的学习和工作中，我将继续努力，不断学习和掌握新的技术，积累更多的项目经验，提高自己的综合素质。我相信，通过不断地实践和学习，我能够在软件开发领域取得更大的进步。 |

**九江学院计算机与大数据科学学院**

## 本科课程实训量化评分标准

学生姓名： 傅森林 指导教师： 王 超 年级和专业： 23级软件工程

|  |  |  |  |
| --- | --- | --- | --- |
| **指标** | **最高分** | **评分要素** | **评分** |
| **设计技术水平** | 30 | 了解企业软件开发过程与规范；核心技术阐述清晰。 |  |
| **实际动手能力** | 30 | 能够完成项目实践，能建立按规范化方法进行软件开发的意识，养成良好的编程习惯，严格的代码规范。 |  |
| **报告规范化** | 20 | 提交的电子文档及打印文档符合规范化要求 |  |
| **回答问题** | 10 | 能简明扼要地阐述设计的主要内容，能准确流利地回答各种问题 |  |
| **学习态度** | 10 | 端正的学习态度及认真刻苦程度等 |  |
| **总 分** | | |  |

**指导教师：**

**年 月 日**

**注意：**

1. 本评分标准适用于计算机软件课程设计；
2. 总分满分为100分，成绩参考标准为：优秀（100>X≥90）；良好（90>X≥80）；中等（80>X≥70）；及格（70>X≥60）；不及格（X<60）；
3. 发现有拷贝舞弊现象者，一律直接退回不作检查，两次舞弊者按不及格处理。
4. 每个学生至少要回答三个以上的问题，有两个以上问题回答不清楚者，一律不及格。
5. 课程设计报告不交者一律不及格。