****

课程实训日志

（ 2024 — 2025 学年第 2 学期）

课程名称： 软件工程课程设计

学 院： 计算机与大数据科学学院

专 业： 软件工程

班 级： 计科A2351

学 号： 23

学生姓名： 傅森林

指导教师： 王超 王江涛

起止时间： 2025年3月3日 至 2025年3月7日

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实践日期 | 2025年3月3日 | | 实践地点 | 电子实验楼506 |
| 实践主要内容   1. **软件公司研发流程与规范介绍**：了解软件公司整体的软件研发流程以及遵循的规范，为后续项目实践奠定基础。 2. **软件工程师岗位职责与要求及职业发展规划介绍**：明确软件工程师岗位的具体职责、要求，并知晓该岗位的职业发展路径，有助于个人规划。 3. **项目介绍与需求分析**：对实训项目进行详细介绍，分析项目的具体需求，使学员清楚项目目标和方向。 4. **GIT项目管理**：学习使用GIT进行项目版本控制和管理，掌握代码的提交、分支管理、合并等操作。 5. **开发环境搭建**：包括安装Python3，以及进行PyCharm的安装和配置，为后续的代码编写和项目开发做好准备。 6. **知识点学习**：    * **GIT项目管理**：深入学习GIT的使用，确保能够熟练运用其进行项目协作和版本管理。    * **Python中的面向对象**：掌握Python中面向对象编程的概念、类和对象的使用、继承、多态等特性。    * **Python中的正则表达式**：学习正则表达式的语法和在Python中的使用方法，用于字符串匹配和处理。 | | | | |
| 存在问题及解决情况   1. **概念理解**    * **软件研发流程与规范**：对敏捷开发迭代流程、规范实际应用理解不深。    * **面向对象编程**：抽象、封装、继承、多态等概念抽象，应用场景和机制难把握。    * **正则表达式**：语法复杂，量词、分组等概念易混淆，编写复杂式子困难。 2. **工具使用**    * **GIT**：分支管理和合并时冲突解决不当。    * **PyCharm**：配置时解释器路径设置错、插件安装失败。 3. **需求分析**：缺乏经验，难全面准确理解项目需求，开发方向易偏差。   **解决措施**   1. **概念理解**    * **查阅资料**：看专业书籍、文档、教程，如《敏捷软件开发：原则、模式与实践》。    * **实践案例**：结合代码和项目案例理解，用正则处理日志等。    * **请教他人**：向导师、同学请教，参加技术交流活动。 2. **工具使用**    * **官方文档**：阅读GIT和PyCharm官方文档找解决方案。    * **模拟练习**：创建模拟项目练习GIT操作，按指南配置PyCharm。    * **社区支持**：在Stack Overflow等社区搜索问题、求助。 3. **需求分析**    * **沟通确认**：与相关人员充分沟通，记录确认关键信息。    * **文档分析**：研读需求文档，拆解任务，做分析报告。    * **参考项目**：借鉴类似项目需求分析和实现方式。 | | | | |
| 备注 | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实践日期 | 2025年3月4日 | | 实践地点 | 电子实验楼506 |
| 实践主要内容   1. **数据爬取与分析功能完成**：在项目里承担对在线招聘网数据的爬取工作，并实现相应的分析功能。这要求能够准确从招聘网站获取所需信息，如职位名称、薪资范围、任职要求等，再对爬取到的数据进行整理和分析，例如统计不同职位的需求数量、薪资分布情况等。 2. **深度优先算法**：可能运用深度优先算法来辅助数据爬取，比如在网页链接的遍历过程中，按照深度优先的顺序探索招聘网站的不同页面，以确保全面获取数据。 3. **知识点学习与应用**    * **Python中的request库的使用**：掌握request库的基本操作，包括发送HTTP请求（如GET、POST请求）来获取网页内容，设置请求头、参数等，以模拟浏览器访问招聘网站，获取数据。    * **Python中的beautiful库的使用**：学习使用beautiful库（应为BeautifulSoup库）来解析HTML或XML文档，从中提取出所需的数据信息，例如通过标签、类名、ID等定位和提取招聘相关信息。 | | | | |
| 存在问题及解决情况   1. **数据爬取**    * **反爬机制**：网站有限制访问频率、验证码等反爬措施，触发后难获取数据，不知如何绕过。    * **网页结构**：网站结构不定期更新，原爬取代码难适应新结构提取数据。    * **算法应用**：用深度优先算法遍历链接，易陷入循环或遗漏重要链接，获取数据效率低。 2. **数据分析**    * **数据清洗**：爬取数据有缺失、重复、格式不规范问题，缺乏清洗方法和经验。    * **方法选择**：统计职位需求、薪资分布等时，难选合适分析和可视化方法呈现结果。 3. **库的使用**    * **request库**：处理设置代理、SSL证书等复杂请求场景有困难。    * **BeautifulSoup库**：解析复杂页面时，难准确定位和提取信息。   **解决措施**   1. **数据爬取**    * **反爬机制**：查资料了解应对方法，如设访问间隔、用代理IP，验证码用打码平台或机器学习识别。    * **网页结构**：编鲁棒性强的代码，用灵活选择器，定期检查结构并调整代码。    * **算法优化**：代码中设终止条件和访问标记，筛选分类链接，优先访问重要链接。 2. **数据分析**    * **数据清洗**：用pandas库处理缺失、重复值和格式转换，如用dropna()删缺失值，drop\_duplicates()去重复值。    * **方法选择**：参考书籍案例选分析方法，用matplotlib、seaborn库可视化，如柱状图展职位需求，箱线图展薪资分布。 3. **库的使用**    * **request库**：查官方文档和社区教程，用proxies设代理IP，verify处理SSL证书。    * **BeautifulSoup库**：深入研究文档，结合选择器和XPath解析，复杂页面分步提取信息。 | | | | |
| 备注 | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实践日期 | 2025年3月5日 | | 实践地点 | 电子实验楼506 |
| 实践主要内容  该实践主要围绕在企业级MySQL数据库环境下进行项目数据库搭建、数据表创建及操作展开，具体内容如下：   1. **数据库环境搭建与配置**：使用企业级MySQL数据库环境，完成数据库的搭建与相关配置工作，为后续的数据库操作提供基础。 2. **数据库与数据表创建**：创建项目数据库，并在该数据库中新建用于存储招聘信息的数据表，设计数据表的结构以适配招聘信息的存储需求。 3. **SQL语句操作**：针对招聘信息表，运用SQL语句进行各类操作，包括但不限于数据的插入、查询、更新和删除等，以实现对招聘信息的有效管理。 4. **数据导入**：将爬取得到的招聘信息数据导入到项目数据库的招聘信息表中，完成数据的持久化存储。   涉及的知识点包括：   1. **SQL语言**：掌握SQL语言的基本语法，理解不同数据类型的特性，熟悉数据表的创建、修改和删除等操作。 2. **CRUD数据操作**：学会使用SQL语句实现数据的增加（Create）、读取（Read）、更新（Update）和删除（Delete）操作，这是数据库操作的核心内容。 3. **Python连接数据库并操作数据**：了解如何使用Python编程语言连接MySQL数据库，并通过Python代码执行SQL语句来操作数据库中的数据，实现数据的自动化处理。 | | | | |
| 存在问题及解决情况  **存在问题**   1. **环境搭建**：企业级MySQL安装复杂，有版本兼容与依赖问题；配置参数多，难理解取值，易配置不合理。 2. **创建设计**：数据表设计欠合理，完整性、扩展性不足；命名规范不统一，影响可读性与维护。 3. **SQL操作**：易出语法错误，如关键字拼写、括号不匹配；复杂查询编写困难。 4. **数据导入**：数据格式与表结构不匹配，有乱码风险；数据量大时导入慢，可能超时。 5. **Python连接**：连接数据库可能失败，如用户名、密码错误或网络问题；代码存在SQL注入安全风险。   **解决情况**   1. **环境搭建**：查官方文档与教程安装，遇问题搜解决方案；参考优化指南调整参数，监控性能优化。 2. **创建设计**：学习设计知识，评估表结构，添加约束、选合适字段类型；制定统一命名规范并共享。 3. **SQL操作**：用工具检查语法，读错误信息修正；学高级教程与案例，练习复杂查询。 4. **数据导入**：用pandas预处理数据，统一格式；批量导入，调整数据库参数提高速度。 5. **Python连接**：检查连接信息与网络；用参数化查询，过滤验证输入防SQL注入。 | | | | |
| 备注 | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实践日期 | 2025年3月6日 | | 实践地点 | 电子实验楼506 |
| 实践主要内容  该实训第4天的实践主要聚焦于在线招聘网数据的爬取、存储、展示以及相关接口测试，同时涉及Python操作数据库、GUI开发、图表绘制和测试工具使用等内容，具体如下：   1. **数据爬取与存储展示功能**：完成项目中爬取数据并保存至数据库的功能，同时实现数据显示功能。这要求从在线招聘网爬取所需数据，如职位详情、公司信息等，并将其存储到数据库中，然后从数据库读取数据并以合适的格式展示出来。 2. **在线招聘网数据爬取及图表显示**：进一步完成在线招聘网数据的爬取工作，并实现图表显示功能。例如，将招聘数据（如不同职位的需求数量、薪资分布等）以图表（柱状图、饼图等）的形式直观呈现。 3. **接口测试**：使用jmeter进行增加数据、显示数据等接口测试。通过jmeter模拟不同的请求，对系统中与数据增加和显示相关的接口进行测试，验证接口的正确性、性能和稳定性。   涉及的知识点包括：   1. **Python读取mysql数据库中数据**：掌握使用Python连接MySQL数据库并执行查询操作，获取所需数据的方法，如使用pymysql等库。 2. **Python中GUI库的使用**：学习使用Python的GUI库（如tkinter等）创建图形用户界面，以便更好地展示数据和与用户进行交互。 3. **echarts图库的运用**：了解echarts图库的基本概念和使用方法，能够将其集成到项目中，实现数据的可视化展示。 4. **jmeter测试工具的使用**：熟悉jmeter测试工具的界面和基本操作，能够创建测试计划、添加采样器、配置断言等，对系统接口进行测试。 | | | | |
| 存在问题及解决情况  **存在问题**   1. **数据操作展示**：爬取数据存数据库及展示时，可能遇数据格式不匹配、存储异常，图表显示数据不准确、样式不佳。 2. **接口测试**：用jmeter测试接口，可能出现请求参数设置错、接口响应异常难定位，测试计划配置复杂易出错。 3. **知识点应用**：Python读MySQL数据可能连接失败、查询语句有误；GUI库使用界面设计不合理、交互响应差；echarts运用图表配置复杂、数据绑定困难。   **解决情况**   1. **数据操作展示**：爬取前规范数据格式，存取时异常捕获处理；调试图表代码，参考案例优化样式。 2. **接口测试**：仔细检查请求参数，用jmeter查看响应分析异常；查阅文档和教程配置测试计划。 3. **知识点应用**：检查Python数据库连接信息与查询语句；参考GUI设计规范优化界面；学习echarts文档配置图表、绑定数据。 | | | | |
| 备注 | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实践日期 | 2025年3月7日 | | 实践地点 | 电子实验楼506 |
| 实践主要内容  该实践主要围绕项目收尾阶段的关键工作展开，具体内容如下：   1. **整合项目与测试**：对项目的各个组成部分进行全面整合，确保不同模块、功能之间能够协同工作。同时，开展系统测试，包括功能测试、性能测试、兼容性测试等，以检验项目是否满足预期的需求和标准，发现并修复潜在的问题和缺陷。 2. **项目PPT等文档的编写**：编写项目PPT，用于清晰、直观地展示项目的背景、目标、实施过程、成果、创新点以及未来展望等内容。此外，可能还需编写其他相关文档，如项目总结报告、技术文档、用户手册等，详细记录项目的相关信息，为项目的交付、使用和维护提供支持。   PPT汇总 | | | | |
| 存在问题及解决情况  **存在问题**   1. **整合测试**：项目整合时模块兼容差，测试易发现功能、性能等问题，定位修复难题多。 2. **文档编写**：PPT及文档内容结构乱，重点不突出，编写耗时，格式规范难统一。 3. **评审答辩**：汇报时紧张致表达不清，对项目细节把握不足，难应对专家质询。   **解决情况**   1. **整合测试**：整合前统一标准，测试用专业工具定位问题，团队协作修复并复测。 2. **文档编写**：规划内容结构，参考模板，用文档工具统一格式，分工编写后审核整合。 3. **评审答辩**：提前演练，熟悉项目细节，准备常见问题答案，答辩时保持冷静自信。 | | | | |
| 备注 | |  | | |