有效方法

实验一：软件需求分析

关键问题：选择将要研究的开源项目；针对选择的开源项目进行需求分析。

解决方法:

1. 选择开源项目

目的：选择小组研究的开源项目。

依据：根据小组内成员的研究方向、知识背景选择待研究的开源项目。

人员：小组所有成员。

处理步骤：小组内成员收集所在实验室的研究方向相关资料，集中进行讨论，最后决定开源项目。

结果：决定出小组研究的开源项目。（Scrapy ）

1. 项目需求分析

目的：进行所选项目的需求分析，以期得到改进的灵感。

依据：在获取Scrapy源码的前提下，参考Scrapy官方文档等资料。

人员：主要编写人为武丁泽宇，其余组员进行修改

解决步骤：首先了解软件的总体框架，然后参考并分析一下Scrapy，最后绘制Scrapy的用例图。

结果：得到Scrapy需求用例图。

实验二：软件需求评审

关键问题：寻求对系统进行改进和扩展的地方；考虑爬虫结果的展示。

解决方法：

1. 寻求对系统进行改进和扩展的方法

目的：通过对现有系统的分析，寻找不足和缺陷，并提出改进方法。

依据：Scrapy源码，实验一得到的Scrapy需求分析和用例图。

人员：全体成员。

解决步骤：重新分析实验一Scrapy的需求，根据Scrapy中某些可以改进的点，提出一些改进方案。

结果：得到改进方案，例如将Scrapy改为分布式爬取，使用反爬机制等。

1. 考虑爬虫结果的展示

目的：分析Scrapy中还有哪些可以改进部分，最后考虑反爬结果的展示。

依据：Scrapy的源码，考虑到Scrapy哪些部分可以改进。

人员：王益飞、郭炜锋、胡勇。

解决步骤：首先分析Scrapy的架构，考虑到暂时不存在改进点。然后由于代码量的不足，在Scrapy项目中不能改进的基础上增加代码量，最后提出以前端网页展示抓取结果。

结果：前端展示顺利的完成，同时满足了代码量。

实验三：改进与展示

关键问题：是修改Scrapy的源码，还是重新实现满足改进需求的功能。

解决方法：

1. 是修改Scrapy的源码，还是重新实现满足改进需求的功能。

目的：针对实验二得到的改进需求，需要落实在代码中。

依据：Scrapy源码，实验二得到的改进需求。

人员：小组所有成员

解决步骤：

1. Scrapy分布式爬虫实现（王益飞）：将scrapy改造成一个多机分布式爬虫，实现多台机器对同一个爬取任务的共同完成。。
2. 反爬（郭炜锋）：使用到的两种基本策略为：伪装user-agent和免费代理。
3. UI展示部分（胡勇）：前端展示Scrapy抓取的结果。

结果：由于抓取网址的原因，反爬机制没有使用上，其他改进都符合预期结果。

实验四：软件测试

关键问题：检查实验三开发的软件是否完成了软件需求规格说明书中所要求的需求，检查所开发的代码是否正确可靠。

解决方法：

1. 目的：测试并记录所开发的软件的测试结果，撰写测试报告。
2. 依据：软件需求规格说明书和测试规格说明书，进行需求完整性测试和实现代码测试。
3. 人员：小组所有成员，武丁泽宇负责文档编写的主要部分
4. 解决步骤：按照我们组员各自实现的部分来设计测试用例
5. Scrapy分布式爬虫实现（王益飞）：编写分布式Scrapy的测试用例以及数据库的测试。
6. 反爬（郭炜锋）：编写伪装user-agent和免费代理的测试以及后端服务器的测试用例。
7. UI展示部分（胡勇）：编写前端网页的测试用例。

实验五：软件测试评审

关键问题：怎么看待同学评审中提出的建议。

解决方法：根据自己实验的情况和能力水平对同学的建议筛选，采纳力所能及的建议。

目的：怎么看待同学评审时提出的所有建议，尤其是一些很有难度的建议。

依据：软件测试时评审建议，自己实验的整体进展情况，自己小组的能力和时间。

人员：小组所有成员

解决步骤：

针对测试评审中同学提的很有水平的建设性建议如何进行筛选，我们的解决方法：

根据自己实验的实际进展情况和小组自身的能力和精力，采纳一些难度适中的改进建议。由于自身能力和投入精力的限制，对测试当中的一些有难度的测试我们进行了舍弃

结果：根据采纳的评审意见对自己的测试文档进行了相应的修改。

实验六：进度计划与控制

关键问题：如何高效整理对老师每周分配的任务和评审建议。

解决方法：按照时间顺序，以每周的实验内容为主题，对建议进行分类整理。

目的：对实验的进度整理的时候，由于提交的文档很杂很多，而且过去了很长时间，对有些问题和记录表达的意思不是很清晰，如何高效的整理这些文档和评审建议。

依据：每次实验的内容，每周课堂记录的评审建议和老师所布置的任务以及小组内部的任务分工。

人员：王益飞进行进度计划的修改和生成进度计划与控制分析报告（主要负责人员），郭炜锋进行课堂记录。

解决步骤：

首先，我们明确每周的主要任务和课堂评审的建议的重点，生成项目计划书以及每周具体的mmp文件

结果：提交项目计划书和持续更新的mmp文件。

实验七：配置管理

关键问题：编写《配置管理总结》的时候，存在大量的草稿、很多文档相似性很高，不确定该如何辨别出每个提交的文档与之前提交的文档的不同之处。

解决方法：针对改进后差别不大的文档，我们对文档的改进进行了更为细致的记录，专门用一个文档按照时间顺序跟踪每个文档的改进情况。

目的：如何辨别github上提交的大量内容差别不大的文档。

依据：每个实验提交的文档及改进后的文档。

人员：全体成员，其中郭炜锋进行整理。

解决步骤：

首先，我们本次实验中直接把每次实验的文档全都写进了《配置管理总结》当中，然后针对以后的文档，我们对每个版本的改进做了一个单独的记录。

结果：提交的《配置管理总结》。

实验八：工作量估计与统计

关键问题：怎么对组员的工作做出客观、公正的量化。

解决方法：小组内经过商量产出每个人的工作的权重。

目的：公正客观的评价组员的工作量。

依据：

平时的表现，如积极程度

分配的任务数，如实现的功能模块数

Github提交文档总数

实现功能的难易程度

每次完成的情况（统计代码行数、文档字数）

人员：全体成员，胡勇进行整理。

解决步骤：

针对如何对组员的工作做出客观、公正的量化问题，我们的解决方法是：

首先全体成员协商一下自己分配的任务，主要是征求每个人的意见，尽量满足每个人的要求。

根据大家的考虑重点和建议做出一个全面的、客观的评价标准。

然后，对每个人的工作量的评价，综合每个人整个实验中的所有工作产出：如github上提交的文档数，编写代码的行数，每次实验提交文档的字数，分配的任务的难易程度，然后每一个评定项都给出一个权重。

结果：提交统计报告。