软件工程综合实验数据分析报告

1. 依据
   1. Lua研究参考资料
2. lua-5.2.4.tar(源码)
3. Lua源码欣赏
   1. 需求阶段
4. Lua-系统需求规格说明书
5. 需求评审单（对其他小组的评审）
   1. 测试阶段
6. Lua-软件测试需求规格说明书
7. 被测软件执行结果分析报告
   1. 管理配置
8. 变更和管理分析报告
9. 工作量分析报告
10. 进度控制分析报告
11. 进展日志
12. 实验6-8计划
    1. 计划安排
13. 开源软件研究计划
    1. 其他
14. 每周PPT
15. 每周问题记录、回答及分析
16. 时间表
17. **实验数据分析目标**
    1. 项目选题方向的重要性分析
    2. 项目进度控制方面的数据分析
    3. 团队成员分工与成员能力优势相关度分析
18. **数据分析要点**
    1. **项目选题方向的重要性分析**
       1. 数据整理

本分析要点的数据来源主要是项目选题之后，在项目过程中所生成的文档。选择其中对于本分析点有用的文档，并且进行了一下删除和总结，主要包括Lua-系统需求规格说明书、Lua-软件测试需求规格说明书、被测软件执行结果分析报告、变更和管理分析报告。

* + 1. 数据统计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文档名称 | 文档所属阶段 | 文档更改次数 | 文档消耗总时间 | 与该文档有关的小组讨论次数 |
| Lua-系统需求规格说明书 | 需求阶段 | 23次（包括小修改） | 36小时 | 5次 |
| Lua-软件测试需求规格说明书 | 测试阶段 | 12次 | 20小时 | 3次 |
| 被测软件执行结果分析报告 | 测试阶段 | 6次 | 10小时 | 3次 |

* + 1. 数据特征及其相关性分析

根据以上数据统计，分析得出的数据特征包括：

1. Lua-系统需求规格说明书，该文挡的文档修改次数和文档消耗时间总时间以及小组讨论次数远多于其他两个文档。
2. Lua-系统需求规格说明书的文档修改次数和消耗总时间不是呈非线性增加。
3. 被测软件执行结果分析报告的文档修改次数和消耗总时间是测试需求规格说明书的大约一半左右。
4. 被测软件执行结果分析报告的小组讨论次数大约与测试说明书持平。

根据以上的数据特征，我们讨论一下这些数据特征与项目选题方向的重要性的相关性。

1. 第一条数据特征反映了项目初期我们对该项目的确切研究方向还是出比较朦胧的阶段，在摸索中前进，所以消耗的时间和小组讨论次数比较多，以及修改比较频繁，但是随着项目的进程，我们组内成员对项目方向的理解日益增加，所以后来消耗的时间越来越少。
2. 第二条数据特征反映了我们相对消耗的总时间，我们的修改次数更加显著，这说明在项目初期我们对文档的修改都是小范围的修改，这说明我们在确定了大方向的前提下，加深理解和具体化方向并不是很消耗时间。
3. 第三条数据特征反映了执行结果报告只是分析过程的实践，抽象的表达为方向比实践更重要。
4. 第四条数据特征反映了执行结果报告对于测试点分析中每个分析点都是相对应的，所以需要相对的讨论次数
   * 1. 分析结论

经过以上的数据分析我们得出了以下结论：项目选题方向的对于整个项目很重要，好的开始是成功的一半，所以在项目选题一定要慎之又慎，并且要多方面考究和讨论。

* 1. **项目进度控制方面的数据分析**
     1. 数据整理

项目进度的数据主要是实验6的数据，包括项目研究计划和项目日志，其中项目研究计划中记录了一些计划和实施情况，该计划已经于6月7号更新至最近的情况。日志则是每周细节的实施信息，也于6月7号同步至最新。

* + 1. 数据统计

由于在实验7中我们已经依据两项数据分析了工作量的多少，我们不再分析每个人的工作量，而是分析整个实验过程安排的合理性，任务计划的合理性以及任务执行过程中出现的状况分析。

整个实验计划分为这些阶段：确定及初步研究，软件需求分析，软件需求评审，软件需求复评审，软件需求完善，软件测试需求分析，软件测试评审，中期实验评审，软件测试复评审，软件演示和项目总结。

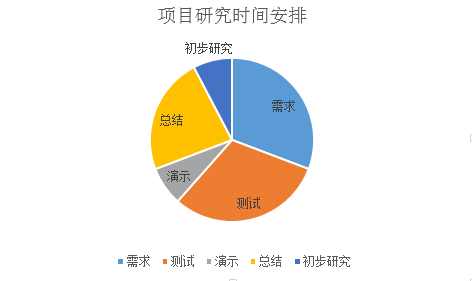
6月10号是开学第14周，也就是软件课的第13周，所以到6月10号为止，我们的研究已经进行了12周。从数据上看，软件需求方面的实验持续了4周，软件测试方面的实验持续了也是4周，而软件演示由于测试的准备只需要1周，项目总结持续3周，但到现在只有2周，还有初步研究的第1周。另外，软件计划和配置方面的实验是贯穿始终的。

从Github上，我们可以看到整个计划修改了7次，第一次可以看到一个计划的基本轮廓，第二次添加了时间结点和资源信息，第三次完善成一个正常的计划，后面几次则是一些重要时间结点的计划和日志更新。后面的更新中，有过期修正计划的情况。

状况包括工作未做和推迟，其中有2次评审工作未做，共9次工作推迟。

* + 1. 数据特征

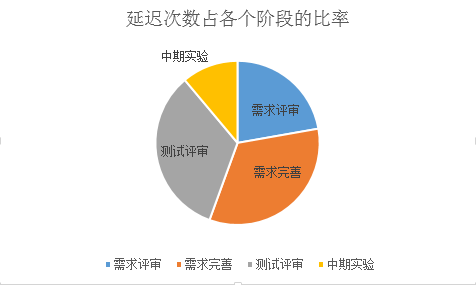
对于整个实验的时间安排，我们大致将时间划分如下，其中由于项目计划的每周更新将占用一些时间，但是在每个阶段几乎是均匀占用的，因此计划方面的内容不列入图表统计。



可以看到需求和测试占用了大部分的时间，这正是我们实验的重点，我们知道初步研究，需求，测试，演示这四个模块，每一个模块都需要前面的模块作前置，所以他们能否研究取决于前面的研究是否进行完。就需求而言，初步研究做前置有一点不合适，初步研究不能做出一个很好的需求分析来，不过需求分析的总体持续时间很长还是弥补了这一缺点。

计划的修改基本符合预期，在统计中发现后面的4次均有计划后补的情况，通常是本周的内容在这周即将结束的时候补上。虽然这样计划的准确性高了一点，但是却有点体现不出计划的领导意义来。

未做的工作一次是软件需求复评审阶段，一次是测试需求评审阶段，第一次没有做的原因是工作失误，第二次没有做的原因是论坛出了问题，第二次的评审在出问题之后，替换成了另外一个评审工作，即评审中期材料。关于推迟现象，我们对9次的追踪记录中发现，需求评审阶段有2个，需求完善阶段有3个，测试需求评审时有3个，中期实验评审时有1个。



需求评审阶段延迟主要是RUCM未完工，需求完善时的推迟主要是计划失误，是比较正常的延迟，测试评审阶段的延迟主要是测试需求说明书未完工问题。中期实验的问题推迟到本周的另外时间，也算合理延迟。可见导致延迟的主要问题是技术还不够熟练。

* + 1. 分析结论

就整体研究过程而言，我们给出的结论是需求前面应当添加一个相对深入的研究阶段，或者分一部分时间到初步研究阶段。

计划应当及时制定，及时跟进。

需求和测试需求分析不熟练是导致我们计划延迟的主要原因。