

**进度计划实验、工作量分析实验**

**及配置管理实验研究计划**

北京航空航天大学

2015-06

小组成员

|  |  |
| --- | --- |
| 名字 | 学号 |
| 黎功辉 | SY1406232 |
| 白瑞雪 | SY1406233 |
| 颜世增 | SY1406219 |

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| 1.0 | 2015/5/13 | 颜世增 | 全体 | 该文档的最初版本 |
| 2.0 | 2015/6/17 | 颜世增 | 颜世增 | 该文档的最终版本 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 软件进度计划与控制 4](#_Toc422320077)

[1.1 计划设计 4](#_Toc422320078)

[1.2 计划跟踪 4](#_Toc422320079)

[1.3 评估完成情况 5](#_Toc422320080)

[1.4 进度控制分析 5](#_Toc422320081)

[2 工作量分析与估计 5](#_Toc422320082)

[2.1 计划工作量 5](#_Toc422320083)

[2.2 实际工作量 6](#_Toc422320084)

[2.3 统计与分析 6](#_Toc422320085)

[2.4 成员分配情况 6](#_Toc422320086)

[3 配置管理 6](#_Toc422320087)

[3.1 github管理 6](#_Toc422320088)

[3.2 变更分析 7](#_Toc422320089)

本文档的编写主要为了指导开源软件研究实验的6-8实验如何做和怎么做的计划，实验6为软件进度计划与控制，实验7为工作量估计与统计分析，实验8为配置管理。软件进度计划与控制主要指的是我们确定的Lua开源软件的研究过程的计划，以及针对计划做的一些控制类文档。工作量的估计与统计一方面总结计划的工作情况，同时结合实际的时间情况，分析计划制定的合理性，以及成员的完工情况。配置管理主要针对的是我们所有制品的管理情况，设计github的版本变更情况，以及版本设计的一些规则等。

# 1 软件进度计划与控制

本实验主要是为每个成员制定合适的任务，以及合理控制每个成员对该任务执行情况的追踪，主要用的是MS Project来制定任务和记录任务完成情况。Project中主要为任务的完成情况对比，细节则记录在每周的工作日志中。

## 1.1 计划设计

我们的大任务设计，是根据老师的每周任务情况设计的，为该周需要进行的一个抽象任务。然后将该任务再细分两级，构成三级式的任务模式。而由一级任务到二级任务的划分主要是对每周大任务的一个逻辑划分，按照时间逻辑或者任务需求逻辑。三级任务则是可以具体到人的细分事项。

所以我们的整体设计是一个树形的任务划分，每个成员只需要关心对应给自己的第三级，也就是叶子节点的任务即可。

设计过程所涉及到的属性包括时间，需求，依赖关系，和个人能力等，同时，也较多依赖于我们组的成员交流及时度。

所涉及的具体结果为开源项目研究计划，每周由任务员完善该周的情况记录和制定下周最新的详细计划。详情见开源项目研究计划.mpp

## 1.2 计划跟踪

计划跟踪情况分两部分，一部分是在Project中记录到目前为止的计划完成情况，未完成或者有拖延完成的都要有记录，并简单的解释原因。而详细的任务情况，则需要到日志中查找。所以我们日志设计是记录每周中，每个人的工作完成情况，若出现异常时，记录本周异常发生的原因。

日志的整体设计也是按照每周的情况来，主要格式是本周的哪一天，谁做了什么事，事情要详细展开，争取做到事后分析时，有理可依。

当完成时间短短少于计划的时间时，小组成员可以报告给计划设计者，设计者在任务设计处做出标记，并由小组讨论时间安排是否确实存在浪费的情况，以为下次做设计时，提供一个合理的时间考虑。

我们的日志成品为用word记录的进展日志。该日志原则上为每天进行维护，但从人性化考虑，我们决定由计划设计者每周记录项目完成情况。详情见进展日志.docx

## 1.3 评估完成情况

由于所有的设计都有一定的制品来控制，我们以制品形式评估结果完成情况。小组成员提交制品后，每个成员进行传阅，并决定需要修改的地方，然后发回给该成员完善，如此反复之，直到小组成员没有意见。

所以本处记录的时间应该是制品最终完成的时间，由于每个成员每天时间的安排自由度不一样，一般来说，在计划当天完成的制品，不论时间早晚，都算在计划的时间里完成了任务。

该实验没有指定惩罚措施，但从本周开始，每个要讲项目的成员为本周的监控者，需要及时监督每个成员的制品完成情况。

## 1.4 进度控制分析

进度控制分析是对当前进度的一个控制和分析，本工作预备在整个研究过程只执行2-3次，第一次是工期中期，在准备工作进行的差不多时，编写文档分析进度情况，在工期快要结束时进行一次分析，有必要的话，在这中间再进行一次分析。主要是分析截止该文档编写时发现的计划问题和计划总体进度等。详情请见进度控制分析报告.docx

# 2 工作量分析与估计

本部分主要是对小组每个成员的工作量做记录和分析，以及每个任务的计划工作量和实际工作量。

## 2.1 计划工作量

计划工作量分两种统计方式，一种是每个成员的计划工作量和每个大任务的计划工作量。由于设计时并没有考虑工作的难度，所以仅仅考虑时间可能对每个成员来说是不公平的，因此我们对工作量的评估使用公式：工作量=开发时间\*开发速度，其中开发速度又由工作的难度系数和个人能力决定，其中我们将平均的难度系数制定为3，个人工作能力的水平平均制定为3，因此每个人的工作量=开发时间\*任务难度系数\*个人能力水平，其中任务难度系数和个人能力水平的评估由小组每个成员共同商议认可得到。工作量的计划根据每个任务具体到小时数，对于不足一小时的部分补充到一小时。

计划时间的制定主要靠两种策略，主要是预测时间加事后修正，事后修正不是代表过了任务的计划时间再修正，而是指老师讲解了具体的下周安排之后，我们在这周修正下周的计划。另外一种则是，根据事实或者经验决定时间，例如上课时间，平均每个小组的时间约为45分钟，我们计划为1个小时，开会时间过长会引起小组成员烦躁，因此一般也只有1-2个小时。

## 2.2 实际工作量

实际工作量根据Project数据来得出，也分为每个成员和每个大任务两种不同的情况。主要分析每个成员的时间花费时间，和每个任务的花费时间，通过这两种数据与计划数据做对比，可以得到成员计划与实际工作的对比情况。

## 2.3 统计与分析

对数据的记录暂时采用表格记录，有必要的话可以做成更直观的饼状图或者折线图等。但是由于成员较少，进行到现在的任务数也不多，所以我们现在还没有做这种统计分析，通过列一个表格数据的方法表达了现在的数据情况。详情请见附件4 工作量分析报告.docx（本工作暂时只有对比结果，没有相应的情况分析，请相应负责人员添加）。

需要补充说明的是，不同工作的复杂度是不同的，不同成员在做同一件事情的时候效率也并可能一样，所以从工时上考虑每个成员耗费的时间有时候还不够。但是我们的基本要求是，每个成员至少应为自己不足的工作做出解释，无论是工作难度过大，或者自己有其他事情等这样那样的原因。工作难度是我们比较大的一个可以松弛自己工作的标准，但是需要得到小组成员的一致认可。

## 2.4 成员分配情况

我们小组成员只有三个人，包括黎功辉，白瑞雪和颜世增。在方向的选择上没有比较多的方向可以去选，黎功辉和白瑞雪学过软件工程课程比较多，而颜世增比较擅长一些文档的编写，因此主要从这两方面考虑，决定我们成员分为技术组和文档组，技术组包括黎功辉和白瑞雪，他们负责研究lua的一些核心技术研究，包括需求测试方面的。文档组包括颜世增，在负责项目计划的同时，做一些文档方面的工作和辅助工作，在需求和测试需要人手时也会参加。

# 3 配置管理

主要分析，对所有制品的一个管理，和实验过程中大事项的变更。

## 3.1 github管理

我们所有的制品到现在都有传到github上，但是现在存在这么几个问题，希望能在之后的管理中得到改善。首先，文件没有一个合理的组织，现在是单纯的一个文件叠加的过程，希望小组成员商量后改善这种情况。第二，我们到现在只由一个人向github上传，尽管这种方式比较不容易出现差错，但是难以体验集体开发的过程。

这一部分的管理我们将写到变更和管理分析报告.docx中。

## 3.2 变更分析

变更分析包括所有制品的版本变更的分析，分析主要包含该变更之前的样子，变更之后的样子，以及过程中主要发生的变化记录。本部分也将写入变更和管理分析报告.docx中。

实际上上述两部分我们可以合并到一起，也就是根据github上的提交文档及说明，来查看变更情况，所以我们按照时间序对github上的内容进行了分析并写到文档中。