|  |
| --- |
| 软件工程实验I组 |
| 总结报告v1.0.0 |
| 对Node-RED工具的消息聚集和图形可视化拓展 |

|  |
| --- |
| 暴明坤、夏欣怡、胡俊涛、张雨濛、叶柏威、刘子渊  2020-6-12 |

**目录**

[1 项目概述 1](#_Toc42855932)

[2 实验准备 项目计划 5](#_Toc42855933)

[2.1 概述 5](#_Toc42855934)

[2.2 数据统计 5](#_Toc42855935)

[2.3 数据分析与说明 6](#_Toc42855936)

[2.4 问题与方法 6](#_Toc42855937)

[2.5 总结 6](#_Toc42855938)

[3 实验一 需求分析 7](#_Toc42855939)

[3.1 概述 7](#_Toc42855940)

[3.2 数据统计 7](#_Toc42855941)

[3.3 数据分析与说明 8](#_Toc42855942)

[3.4 文档质量 8](#_Toc42855943)

[3.5 问题与方法 9](#_Toc42855944)

[3.6 总结 9](#_Toc42855945)

[4 实验二 需求评审 10](#_Toc42855946)

[4.1 概述 10](#_Toc42855947)

[4.2 数据统计 10](#_Toc42855948)

[4.3 数据分析与说明 14](#_Toc42855949)

[4.3.1 获取来源 14](#_Toc42855950)

[4.3.2 数据分析 14](#_Toc42855951)

[4.4 问题与方法 15](#_Toc42855952)

[4.4.1 如何进行评审过程，保证评审质量 15](#_Toc42855953)

[4.4.2 如何规范评审过程 15](#_Toc42855954)

[4.5 总结 16](#_Toc42855955)

[5 实验三 设计实现 16](#_Toc42855956)

[5.1 概述 16](#_Toc42855957)

[5.2 数据统计 16](#_Toc42855958)

[5.3 数据分析与说明 17](#_Toc42855959)

[5.4 问题与方法 18](#_Toc42855960)

[5.4.1 如何进行高效的协作开发 18](#_Toc42855961)

[5.5 总结 18](#_Toc42855962)

[6 实验四 测试需求分析 19](#_Toc42855963)

[6.1 概述 19](#_Toc42855964)

[6.2 数据统计 19](#_Toc42855965)

[6.3 数据分析与说明 21](#_Toc42855966)

[6.4 问题与方法 21](#_Toc42855967)

[6.4.1 如何进行有效的测试 21](#_Toc42855968)

[6.5 总结 22](#_Toc42855969)

[7 实验五 测试评审 22](#_Toc42855970)

[7.1 概述 22](#_Toc42855971)

[7.2 数据统计 23](#_Toc42855972)

[7.3 数据分析与说明 26](#_Toc42855973)

[7.3.1 数据来源 26](#_Toc42855974)

[7.3.2 数据分析 26](#_Toc42855975)

[7.4 问题与方法 27](#_Toc42855976)

[7.4.1 如何进行评审过程，保证评审质量 27](#_Toc42855977)

[7.4.2 如何规范评审过程 27](#_Toc42855978)

[7.5 总结 28](#_Toc42855979)

[8 实验六 项目计划与监控 28](#_Toc42855980)

[8.1 概述 28](#_Toc42855981)

[8.2 数据统计 28](#_Toc42855982)

[8.3 数据分析与说明 29](#_Toc42855983)

[8.4 问题与方法 30](#_Toc42855984)

[8.5 总结 31](#_Toc42855985)

[9 实验七 配置管理 31](#_Toc42855986)

[9.1 概述 31](#_Toc42855987)

[9.2 数据统计 32](#_Toc42855988)

[9.3 数据分析与说明 33](#_Toc42855989)

[9.4 问题与方法 33](#_Toc42855990)

[9.5 总结 33](#_Toc42855991)

[10 实验八 统计分析 34](#_Toc42855992)

[10.1 概述 34](#_Toc42855993)

[10.2 数据统计 34](#_Toc42855994)

[10.3 数据分析与说明 35](#_Toc42855995)

[10.4 问题与方法 35](#_Toc42855996)

[10.5 总结 36](#_Toc42855997)

[11 课程总结 36](#_Toc42855998)

# 项目概述

**“基于Node-RED的消息聚集和图形可视化拓展”项目自2020年3月6日启动，历经长达16周的开发周期，至今为止已创建了221个任务（含目录），开发总时长为493.8个工时。项目中工时及任务的燃尽图如图1所示，由该图可以看出，本项目的计划与实际进展较为贴合，项目整体进展顺利。对该图的详细分析可参考第八章。**

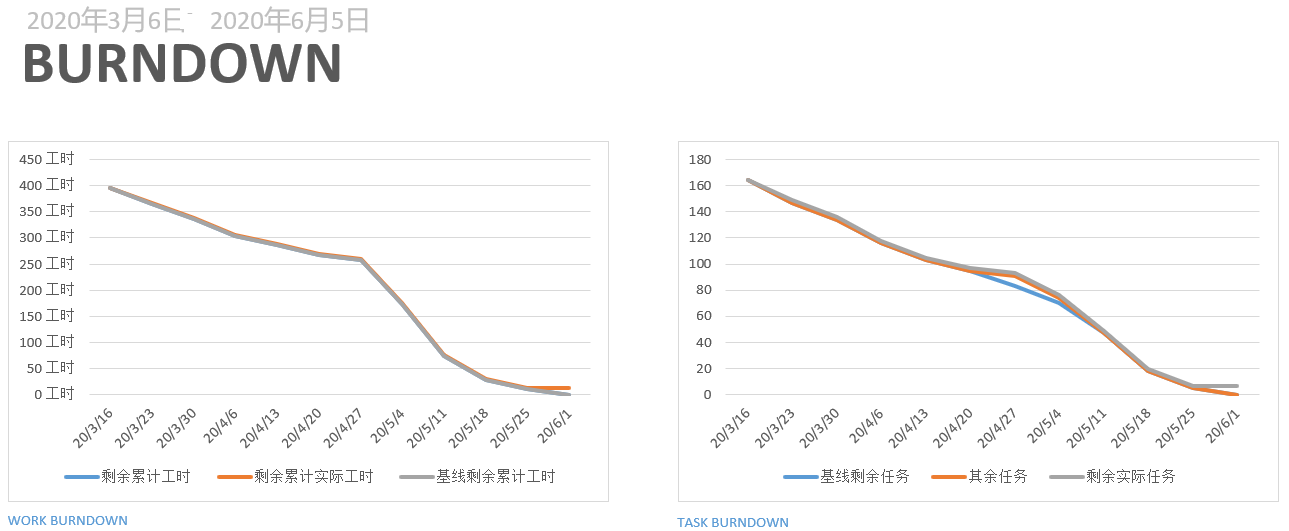


图1 工时及任务燃尽图

小组严格按照软件工程综合实验的要求进行阶段划分，以实验课的内容为依据，将项目分为实验准备、实验一至实验八等阶段来进行。将小组的工作时间按照实验内容进行划分，如图2所示。以阶段划分的角度来看，各个已完成的阶段的预计工时与实际工时十分吻合，表明本组的计划制定地科学合理、行之有效。

图2 各阶段的工时统计

在整个实验过程中，小组成员个人的工时统计以及工时分布情况如图3及图4所示。组内成员所长各有不同，因而在项目进行过程中扮演不同的角色，而项目不同阶段所花费的时间具有较大的差异，小组各成员在项目上最终花费的时间较为接近，表明了组内成员进行了高效的合作，每个人都发挥了自己的长处。**对该图的详细分析可参考第十章。**

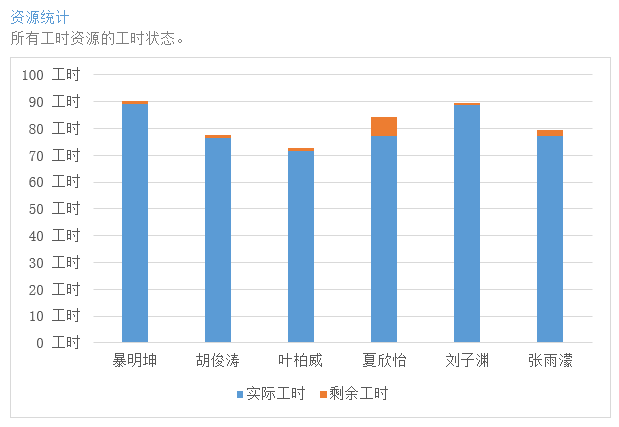


图3 小组成员工时统计

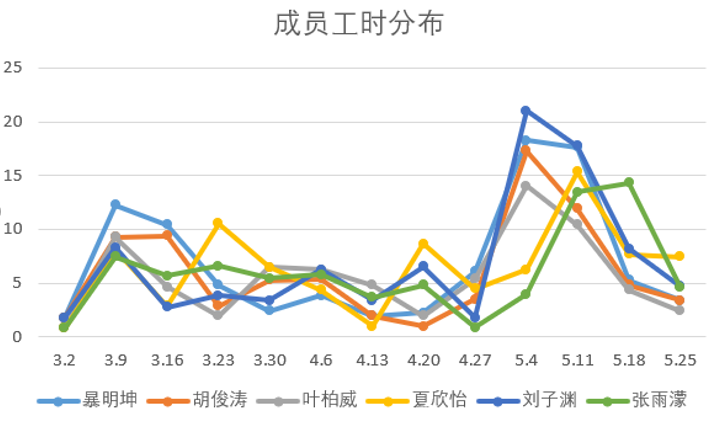


图4 小组成员工时分布情况统计

结合小组成员的工时统计情况，另选取项目进展过程中所编写的文档字数、图表数、需求/测试用例数、评审意见数、代码行数及展示情况等指标，可以得到小组成员的工作量评估情况，如表1所示。

表1 小组成员工作量评估情况

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **文档字数** | **图表数** | **用例个数** | **评审意见数** | **代码行数** | **展示参与数** | **工时统计** |
| **暴明坤** | 7055 | 50 | 33 | 3 | 3307 | 5 | 90.16h |
| **胡俊涛** | 12436 | 48 | 36 | 30 | 1941 | 3 | 77.6h |
| **叶柏威** | 8033 | 49 | 23 | 45 | 1916 | 4 | 72.68h |
| **夏欣怡** | 7610 | 40 | 5 | 35 | 100 | 5 | 79.68h |
| **刘子渊** | 12862 | 57 | 55 | 54 | 1966 | 4.5 | 89.6h |
| **张雨濛** | 14226 | 14 | 5 | 31 | 100 | 2.5 | 78.32h |

在成员贡献率上，由于在开发过程中我们又分为了三个小组，我们决定采用组间互评+组内互评的方式，以工作量评估统计为参考，对不同小组及组内成员进行评分，最终组间的互评结果及成员贡献率如图5及图6所示。

图5 组间互评结果

图6 成员贡献比例图

课程实验自开始以来，历经16周左右，先后进行了八次实验（不包括开始的实验准备及最后的展品展示）。在实验进行过程当中，小组成员积极配合并针对课程进行了时间上的协调，整体配合良好。当然，每一次的实验都有任健老师的课上指导与其他小组成员的帮助，因此，项目在进行当中才能得到不断地修正，最终圆满完成了课程任务。针对整个课程，具体的阶段分析将会在后续章节进行详细展开。

# 实验准备 项目计划

## 概述

**实验准备（即项目计划）的起止时间为2020年3月6日至2020年3月24日，总共历时19个工作日，花费工时为90.8工时。在该阶段，小组的主要工作有明确项目内容、协商工作方式并对项目相关内容进行调研，其中主要产出物包含项目计划书等。**

## 数据统计

**该阶段数据统计如表2所示。**

表2 项目计划书人员数据统计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| E0：软件项目计划 | 软件项目计划工作量统计 | | | |
| 姓名 | 图表数 | 文档字数 | 具体工作内容 | 总耗时 |
| 刘子渊 | 0 | 400 | 修订项目背景与项目目的的介绍逻辑  修订项目目标 | 1h |
| 张雨濛 | 2 | 1000 | 编写项目应交付成果、实施计划等，补充成员分工部分 | 2h |
| 胡俊涛 | 0 | 1500 | 商定调整项目目标  编写第一章引言  导出word并调整格式 | 4h |
| 叶柏威 | 0.5 | 600 | 参与编写引言，修订项目概述 | 1h |
| 暴明坤 | 1 | 1000 | 参与编写引言  编写关键问题  编写项目团队组织 | 2h |
| 夏欣怡 | 1 | 50 | 加入版本修订表 | 0.5h |

## 数据分析与说明

**数据来源于项目计划书的撰写。通过共享文档填写的方式进行收集。该阶段的成员分工与工作内容如上表所示。其中夏欣怡同学由于在设计实验6-8，在该部分参与较少，其余同学各自完成了所负责的工作。**

## 问题与方法

**在项目计划阶段，主要的问题是保证项目开发的可行性。处于保证可行性的第一步，需要了解小组成员的开发能力，明确每个人的技术栈和学习能力。并以此为依据对项目进行初步的选择。对于保证可行性的第二步，需要对项目的规模有一个相对清晰的评估，如果项目的复杂程度太高，或项目本身太过简单，会导致开发无法进行或工作量过少。可行性的第三步是可选项，通过对项目本身的开发基础设施进行评估，如果项目有配套的自动测试等框架，将大大提高开发效率，节省人力物力。**

## 总结

**项目计划阶段决定了整个综合实验的整体走向。我们组在进行项目计划时，计划的相对周密和保守，选择了绝大多数人都能开发的node.js技术栈。同时由于我们的慎重选择，node-red项目本身具有的测试基础设施在后来的测试开发中提供相当巨大的帮助。成为了我们进行自动化测试的基础。**

# 实验一 需求分析

## 概述

**实验一（即需求分析）的起止时间为2020年3月21日至2020年4月3日，总共历时14个工作日，花费工时为43.84工时（不包括评审后文档的修改时间），其中主要产出物包含需求规格说明书。**

## 数据统计

**该阶段数据统计如表3-4所示。**

表3 软件需求分析人员数据统计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E1：软件需求分析 | 软件需求分析工作量统计 | | | | |
| 姓名 | 需求个数 | 图表数 | 文档字数 | 具体工作内容 | 总耗时 |
| 暴明坤 | 8 | 12 | 2000 | 讨论消息聚集的需求建模，完成原理验证 | 3h |
| 刘子渊 | 2 | 4 | 200 | 增加钉钉相关需求详表 | 2h |
| 胡俊涛 | 9 | 5 | 250 | 讨论部分需求的建模；  完成总体功能需求用例图和用例表； | 3h |
| 叶柏威 | 6 | 4 | 800 | 讨论可视化节点需求；  参与编写总体概述、非功能型需求，补充可视化需求用例表 | 2h |
| 张雨濛 | 0 | 1 | 3000 | 搭建需求规格说明书框架；编写引言、业务需求部分；导出word并调整格式； | 4h |
| 夏欣怡 | 0 | 1 | 50 | 加入版本修订表 | 0.5h |

表4 软件需求文档数据统计

|  |  |  |  |
| --- | --- | --- | --- |
| 文档规模 | 需求项数 | 需求类型 | 版本次数 |
| 13183 | 25 | 功能需求（7） | 20 |
| 拓展需求（13） |
| 非功能需求（5） |

## 数据分析与说明

**数据来源于软件需求规格说明书的撰写。通过共享文档填写的方式进行收集。该阶段的成员分工与工作内容如上表所示。在工作量方面，其中夏欣怡同学由于在设计实验6-8，在该部分参与较少，其余同学各自完成了所负责的工作。**

**历经两次需求评审，需求规格说明书最终字数13183字，提出了25项需求，包括7项功能需求，13项拓展需求，5项非功能需求，共计修订版本20次。其中拓展需求及非功能需求为本项目待开发部分，基本涵盖了项目本身所预期的目的，较为全面。**

## 文档质量

**针对软件需求规格说明书的初稿，组内先进行了自行修改，使之尽量符合国标。在课堂上，针对任健老师提出的建议，修改了参考文献的格式与整体框架的规划，提高了文档质量。**

**当然，预期的需求在实际开发过程中会不断进行修改，以使之更加符合用户的使用预期，因此软件需求规格说明书的质量不断提高，内容不断翔实。**

## 问题与方法

**在软件需求分析阶段，需要制定一套科学合理的方法来发掘需求、改善需求。在此过程当中，小组成员通过面对面的多次会议，激烈讨论后最终找到一个令所有成员都可接受的折衷结果。**

**组内成员分别利用在线文档，进行多用户共同在线编辑，方便及时反映不同成员意见和要求，并进行及时记录；通过对往年文档的分析与汇总，归纳包含本组特色的文档格式；对于专业的图表，则以有经验的小组成员为主力，其余成员进行审查的形式保证工作质量和工作效率；为方便其余小组对文档的审查，在文档中加入了较多的说明与对应图表、示例图等，提高了文档易读性；最后，对于文档进行一定的留白，方便日后的修改与扩展。**

## 总结

**需求分析以及需求规格说明书的撰写是小组成立后所面临的第一个实验，具有实验周期短，小组成员没有很好地磨合等挑战，因此文档的初稿存在很多问题，如格式不规范、内容太宽泛等。但是随着评审的进行与组内成员的学习，需求分析阶段得以顺利进行。**

**总的来说，本组的需求分析主要制品——需求规格说明书，具有需求明确、逻辑清晰、可实现性较高的优点，同时也具有篇幅较短的不足。因此，在需求分析时，应该扬长避短，严格遵守相关标准，从用户的角度来分析，而非从产品设计者的角度入手考虑。对于文档撰写，负责人应当针对成员的不同特点来分配任务，以保证高质高量地完成，避免返工。**

# 实验二 需求评审

## 概述

**实验二（即需求评审与复评审）的起止时间为2020年4月4日至2020年4月24日，总共历时21个工作日，花费工时为57.24工时，其中主要产出物包含包括需求文档检查单、对ABGH组的问题清单、对ABGH组的问题反馈文档以及根据评审意见修改后的需求规格说明书。**

**在本实验中，我们小组根据国家标准的需求规格说明书编写规范制定了本组的评审标准（检查单），对ABGH各组的需求规格说明书进行了审查，共产生了4份问题清单，发现了116个问题，其中严重性为“严重”的有9个，严重性为“中等”的有37个，严重性为“轻微”的有70个。**

**与此同时，我们也收到了来自ABGH四组的问题清单，共4份文档，107个问题。针对以上问题我们组均作出了回应并产生了新版的需求规格说明书，其中接受及部分接受问题共91个，拒绝16个。**

**对于已经接受的问题，本组均进行了对应的修改，并产生了新版本的需求规格说明书，此外本组对需求所作的回应以及修改均已被ABGH组确认。**

## 数据统计

需求评审阶段的数据及产出统计如表5-6所示。

表5 本组提出的问题清单统计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 本组产出的问题清单数 | 本组提出的问题总数 | 严重性为“严重”的问题总数 | 严重性为“中等”的问题总数 | 严重性为“轻微”的问题总数 |
| 4 | 116 | 9（7.8%） | 37（31.9%） | 70（60.3%） |

表6 别组提出的问题清单统计

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 别组产出的问题清单数 | 别组提出的问题总数 | | | 严重性为“中等”以上的问题总数 | | | 本组接受及部分接受的问题数以及占比 | | |
| 4 | 107 | A组 | 15 | 25 | A组 | 0（0%） | 91 | A组 | 10（66.7%） |
| B组 | 31 | B组 | 13（41.9%） | B组 | 24（77.4%） |
| G组 | 20 | G组 | 5（25.0%） | G组 | 17（85.0%） |
| H组 | 41 | H组 | 7（17.1%） | H组 | 40（97.6%） |

**该阶段数据统计如表7-11所示。**

表7 本组需求评审人员数据统计（1）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| E2：需求评审 | 需求评审表格设计 3.28-4.3 | | | | | |
| 姓名 | 需求变更 | 图表数 | 报告字数 | 具体工作内容 | 修订日期 | 总耗时 |
| 张雨濛 | 0 | 1 | 500 | 设计评审单 | 4.2 | 1h |

表8 本组需求评审人员数据统计（2）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E2：需求评审 | 评审工作量统计 4.6-4.8 | | | | |
| 姓名 | 评审组别 | 评审意见条数 | 意见字数 | 负责范围 | 总耗时 |
| 胡俊涛 | G | / | / | H组的文档用例、图与RUCM设计 | 1 |
| H | 12 | 999 |
| 刘子渊 | G | 6 | 321 | 非功能需求和运行环境 | 1 |
| H | 6 | 440 |
| 张雨濛 | G | 4 | 128 | 文档格式 | 1 |
| H | 1 | 38 |
| 叶柏威 | G | 18 | 991 | G组的文档用例、图与RUCM设计 | 1 |
| H | / | / |
| 暴明坤 | G | 1 | 44 | 非功能需求和运行环境 |  |
| H | 1 | 46 |
| 夏欣怡 | G | 13 | 644 | 文档语言逻辑与表述，包括语病、错别字、段落逻辑等 | 1.5 |
| H | 4 | 226 |

表9 本组需求评审人员数据统计（3）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E2：需求评审 | 评审修改工作量统计 4.8-4.13 | | | | |
| 姓名 | 修改意见 | 修改图数 | 修改字数 | 具体工作内容 | 总耗时 |
| 胡俊涛 | 6 | 1图7表 | 100 | 修改功能需求用例表，更新拓展用例图 | 2.5 |
| 刘子渊 | 1 | 2表 | 100 | 修改功能需求用例表 | 1 |
| 张雨濛 | 8 | 2表 | 100 | 修改需求规格说明书格式上的一些问题 | 1 |
| 叶柏威 | 23 | 11表 | 200 | 修改格式、用例表、语言逻辑问题 | 1.5 |
| 暴明坤 | 10 | 12表 | 200 | 修改用例表，格式 | 1.5 |

表10 本组需求评审人员数据统计（4）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E2：需求评审 | 评审工作量统计 4.13-4.15 | | | | |
| 姓名 | 评审组别 | 评审意见条数 | 意见字数 | 负责范围 | 总耗时 |
| 胡俊涛 | A | 9 | 437 | A组的文档用例、图与RUCM设计 | 1 |
| B | / | / |
| 刘子渊 | A | 8 | 438 | 非功能需求和运行环境 | 1 |
| B | 6 | 257 |
| 张雨濛 | A | 3 | 117 | 文档格式，段落、标题、引用等 | 1 |
| B | 1 | 43 |
| 叶柏威 | A | / | / | G组的文档用例、图与RUCM设计 | 1 |
| B | 7 | 342 |
| 暴明坤 | A | / | / | 非功能需求和运行环境 | 1 |
| B | / | / |
| 夏欣怡 | A | 8 | 403 | 文档语言逻辑与表述，包括语病、错别字、段落逻辑等 | 1 |
| B | 7 | 468 |

表11 本组需求评审人员数据统计（5）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E2：需求评审 | 评审修改工作量统计 4.15-4.18 | | | | |
| 姓名 | 修改意见 | 修改图数 | 修改字数 | 具体工作内容 | 总耗时 |
| 张雨濛 | 14 | 2表 | 100 | 修改需求规格说明书格式上的一些问题 | 1 |
| 叶柏威 | 33 | 12表 | 200 | 修改语法、格式、用例表、逻辑等问题 | 2.5 |
| 暴明坤 | 3 | 3表 | 50 | 修改语法，格式，用例表 | 1 |

## 数据分析与说明

* + 1. 获取来源

**评审数据来源于ABGH组，表单保存在对应组别的相应目录下，由于问题反馈文档中包含问题清单中的所有内容，所以本组没有进行二次传输。**

* + 1. 数据分析

**需求评审阶段包括需求评审与复评审。在两次评审他组中，按照我们组内的小组划分，三个小组分别主要负责了对非功能需求和运行环境，文档用例、图与RUCM设计与文档语言逻辑表述这三个方面的审查。小组内部自行对两人工作再进行划分。在具体的工作中，暴明坤同学由于特殊情况，在该阶段参与较少，其余组员参与程度较为平均。**

**在评审反馈与修改方面，根据他组提的意见，负责文档相应部分撰写的同学对需求规格说明书进行了修订。夏欣怡同学由于在需求规格说明书撰写中没有很多参与，故没有相应需要修改的部分。其余组员参与程度较为平均，完成了各自的任务。**

## 问题与方法

* + 1. 如何进行评审过程，保证评审质量

**在需求评审过程中，最重要的问题就是如何保证评审质量。由于本组有六名成员要先后对ABGH四个组进行需求评审，该过程中存在多种可选择的评审方案。考虑到时间与质量的权衡，本组最终选择了模块负责与分组并行的评审方案。**

**模块负责是指，对于如项目环境、非功能需求及文档语义和格式等方面，将每人承担一个方面，每周评审两个组，先后对四个组的需求文档进行审查。而对于像功能需求用例描述等较为复杂的方面，将由熟悉该方面内容的同学承担，每周评审一个组，先后两个组的需求文档，然后在小组会议中进行交流，产生最终的评审结果。**

**该评审方案，保证了小组内的每个成员都参与到需求评审中，保证了最终评审结果的可靠性。此外，由于分组并行展开工作，之后再通过讨论的方式快速汇总评审结果，大大提高了需求评审的速度。最后，由于最终产生的问题清单均由全部小组成员检查，也保证了最终评审结果的正确性、有效性。**

* + 1. 如何规范评审过程

**在指定了本组的评审方案之后，组内要考虑如何规范评审过程。由于本组采用两个小组并行展开的方式，所以需要制定统一的评审标准——检查单。**

**本组的检查单根据国家标准（GJB 438B-2009）的软件需求规格说明制定而成，主要从组织和完整性、正确性、合理性和其他等方面共16点对需求文档进行评审。在后续的需求评审过程中，小组成员都以该检查单为标准，进行评审工作。**

## 总结

**在本次实验中，通过不同组别之间的互相评审，我们切实的感受到本组以及别组的文档质量得到了很大的提升。从格式到内容，不止是别组或是老师对我们提出的问题对我们受益颇深，通过整个需求评审过程，我们也更能理解需求的分析与描述方式，更能理解功能需求、拓展需求以及非功能需求之间的区别。**

**在需求评审的过程中，我们共对文档进行了9次修改，产生了对应的9个版本的需求规格说明书。经过这一过程，我们提出了很多问题，也修改了很多问题，文档质量得到了很大的提高。此外，也对我们的项目本身有了更深入的思考，尤其认识到了需求评审阶段的不可或缺。**

# 实验三 设计实现

## 概述

**实验三（即设计实现）的起止时间为2020年4月25日至2020年5月15日，总共历时21个工作日，花费工时为95.08工时。设计实现工作主要分为两个过程，一为根据前期对Node-RED框架内拓展开发方式的调研和尝试以及实验一中对本次项目所需要实现的功能分析，分析Node-RED框架的整体设计并给出本项目的拓展功能实现的具体设计，二为具体代码实现过程，并根据实现中遇到的现实问题对需求和设计进行适当调整。**

## 数据统计

**该阶段数据统计如表12所示。**

表12 软件产品开发人员数据统计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| E3：软件产品开发 | 软件产品开发工作量统计 | | | |
| 姓名 | 代码行数 | 具体工作内容 | 总耗时 | 难度评估/备注 |
| 胡俊涛 | 948 | window、field-join、template-render、cpu-usage、mem-usage | 10h | 一般 |
| 刘子渊 | 645 | feishu-target-remove, feishu-target-add | 18h | 一般 |
| 张雨濛 | 100 | 串联节点，完成demo开发 | 2h | 一般 |
| 叶柏威 | 926 | chart-render | 10h | 一般 |
| 暴明坤 | 1134 | Feishu-activate feishu-msg feishu-send feishu-upload-image feishu-fetch-image feishu-target feishu-all-group feishu-config | 25h | 一般 |
| 夏欣怡 | 100 | 串联节点，完成demo开发 | 5h | 一般 |

## 数据分析与说明

**数据来源于组员对相关节点的开发以及示例的开发，包括代码行数、具体工作内容与总耗时。其中消息聚集组（暴明坤、刘子渊）与数据可视化组（胡俊涛、叶柏威）主要负责相关节点开发，文档与示例开发组（夏欣怡、张雨濛）负责示例的开发。由于node-red应用开发特性，示例开发实际所需编写的代码很少，主要为节点间的数据转化与逻辑串联函数，示例开发部分的代码行数并不能完全反应工作量。在工作量方面，两个软件开发小组在该阶段做了更多工作，完成了节点开发。示例开发组由夏欣怡负责飞书相关节点使用与整体搭建，张雨濛负责图表相关节点与CPU图定时生成。总体而言，组员根据分工，完成了各自工作。**

## 问题与方法

* + 1. 如何进行高效的协作开发

**在代码实现的过程中，协作开发方式的确定至关重要，低效的协作开发成员间任务互相交叠、干涉和冗余，给总体目标的协调和实现带来很大阻力，而高效的协作开发使得成员可以专注于自己任务的同时，又通过适当的交流与其他人沟通完成整体功能的实现。**

**因此根据Node-RED本身拓展节点的开发方式以及本项目的功能需求，将组员分为了三组：消息聚集组、数据可视化组和文档与示例开发组。每个组内的功能高度相关，在每组只有两人的情况下可以通过点对点沟通和结对编程进行高效的开发工作，而组间的依赖则比较轻微，主要集中在示例开发组需要消息聚集组和数据可视化组开发完成之后才能搭建示例，但这也是逻辑上的必须，总的来说遵循了高内聚低耦合的原则。不同组的工作通过git submodule整合到完整项目的目录中，并且各自形成一个Node-RED安装包，在进行集成测试的时候，只需要分别执行命令安装，即可在同一个Node-RED框架中运行该项目开发的所有拓展节点，并且不同组的节点将运行中可能出现的错误本地化并反馈到前端，使得debug过程更快速准确。**

**在组内进行开发的过程中，出现新需求、第三方平台限制或者工期有限的情况，由于不同组对应的需求点也是分立的，可以直接对本组对应的功能需求进行增加和删除，改动波及其他组的状况几乎不存在，也增强了整体的灵活性。**

## 总结

**设计实现是一款产品真正从想法中落到现实中的过程，也是存在问题和挑战最多的地方。我们首先通过对Node-RED框架的整体了解和依照需求分析阶段的工作内容制定了功能实现的方案设计，并按照功能划分为三组形成高内聚低耦合的协作开发方式，再进行密集的代码开发过程，在开发中遇到新想法和现实问题时及时与全组反馈，同时由于低耦合的特点可以灵活便捷地调整原始需求和设计。**

**通过总行数约3600（不包括中间改动的过程）的工作，和依据开发过程中的现实反馈对原有需求和设计的灵活调整，组员们对Node-RED的理解和使用，以及相关的编程数量度都得到了不小的提升，并且对敏捷开发、高效协作有了更好的理解。**

# 实验四 测试需求分析

## 概述

**实验四（即设计实现）的起止时间为2020年5月4日至2020年5月15日，总共历时12个工作日，花费工时为122.88工时（不包括评审后文档的修改时间），其中主要产出物包含测试需求规格说明书、测试报告和软件问题报告。**

## 数据统计

测试需求分析阶段的数据及产出统计如表13-14所示。

表13 测试需求数据统计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文档制品字数 | 测试需求用例个数 | 测试类型 | 测试需求文档更新版本 | 测试结果 |
| 31713 | 76 | 拓展需求测试（67）  场景测试（6）  非功能需求测试（3） | 23 | 全部通过 |

表14 测试需求实验制品统计表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 文件名 | 最后提交时间 | 版本号 |
| 1 | 测试需求规格说明书 | 2019/05/29 | v2.0.5 |
| 2 | 测试报告 | 2019/05/22 | v1.0.1 |
| 3 | 软件问题报告 | 2019/05/22 | v1.0.1 |

**该阶段数据统计如表15-17所示。**

表15 本组测试阶段人员数据统计（1）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E4：测试需求分析与自测 | 测试需求分析 | | | | |
| 姓名 | 测试用例个数 | 测试代码行数 | 测试文档字数 | 总耗时 | 难度评估/备注 |
| 胡俊涛 | 32 | 993 | 7600 | 5h | 一般 |
| 刘子渊 | 51 | 1321 | 7955 | 12h | 一般 |
| 张雨濛 | 5 | 0 | 6000 | 8h | 一般 |
| 叶柏威 | 21 | 990 | 3000 | 5h | 一般 |
| 暴明坤 | 22 | 2173 | 2800 | 18h | 一般 |
| 夏欣怡 | 5 | 0 | 2530 | 2h | 一般 |

表16 本组测试阶段人员数据统计（2）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E4：测试报告 | 测试报告 | | | | |
| 姓名 | 图表数 | 报告字数 | 具体工作内容 | 总耗时 | 难度评估/备注 |
| 张雨濛 | 1 | 1500 | 测试报告初稿 | 3h | 一般 |

表17 本组测试阶段人员数据统计（3）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E4：软件问题报告 | 软件问题报告 | | | | |
| 姓名 | 图表数 | 报告字数 | 具体工作内容 | 总耗时 | 难度评估/备注 |
| 张雨濛 | 1 | 3000 | 软件问题报告初稿 | 2h | 一般 |

## 数据分析与说明

**测试实验部分产出的文件种类较多。各类文件分别有其自身存在的作用。《测试需求规格说明书》总领了整个软件测试阶段的测试规范和测试方向。依据《测试需求规格说明书》进行组内测试，产出了《测试报告》，并根据测试结果进一步撰写了《软件问题报告》。**

**在该实验阶段，组员依据软件开发的分工，完成了各自部分的测试与测试需求规格说明书撰写。张雨濛同学负责了需求规格说明书模板与初稿，并根据测试情况完成了测试报告与软件问题报告文档。总体而言，成员根据分工完成了各自任务，其中张雨濛同学在该部分做了更多的工作。**

## 问题与方法

* + 1. 如何进行有效的测试

**在测试阶段，要注意测试范围和测试规则的指定。测试本身就是一个验证设计开发阶段的成果本身是否产生bug，以及对比开发成果和需求分析中的内容是否产生严重偏差的矫正阶段。由此，测试用例中，对测试规范的制定很关键。**

## 总结

**测试需求阶段，最主要的是将设计实现的成果和需求分析阶段的结果进行总结对比，并对开发成果本身是否出现差错进行验证。本项目的产品从测试结果来看，各项功能的开发成果达到了预期，各个功能模块之间能够正常的协作完成工作，模块内部能够正常实现功能。**

# 实验五 测试评审

## 概述

**实验五（即测试评审）的起止时间为2020年5月12日至2020年5月29日，总共历时17个工作日，花费工时为81.14工时，其中主要产出物包含包括测试需求文档检查单、对CDEF组的文档问题清单及软件问题清单、对CDEF组的问题反馈文档以及根据评审意见修改后的测试需求规格说明书。**

**在本实验中，我们小组制定了本组的评审标准（检查单），对CDEF各组的测试需求规格说明书以及软件程序实现进行了审查，共产生了8份问题清单（包括4份文档问题清单，4份程序问题清单），发现了50个文档问题，33个程序问题，其中严重性为“严重”的有5个，严重性为“中等”的有37个，严重性为“轻微”的有41个。**

**与此同时，我们也收到了来自CDEF两组的问题清单，共8份文档，其中包含57个文档问题，0个程序问题。针对以上问题我们组均作出了回应并产生了新版的测试需求规格说明书、测试报告及软件问题报告。此外本组对测试所作的回应以及修改均已被CDEF组确认。**

## 数据统计

测试评审阶段的数据及产出统计如表18-20所示。

表18 本组提出的问题清单统计

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 本组产出的问题清单数 | 本组提出的问题总数 | 本组提出的文档问题总数 | 本组提出的程序问题总数 | 严重性为“严重”的问题总数 | 严重性为“中等”的问题总数 | 严重性为“轻微”的问题总数 |
| 4 | 83 | 50 | 33 | 5（6.0%） | 37（44.6%） | 41（49.4%） |

表19 别组对我组测试文档提出的问题清单统计

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 别组产出的问题清单数 | 别组提出的问题总数 | | | 严重性为“中”以上的问题总数 | | | 本组接受并修改的问题数以及占比 | | |
| 4 | 57 | C组 | 18 | 21 | C组 | 7（38.9%） | 44 | C组 | 15（83.3%） |
| D组 | 18 | D组 | 4（22.2%） | D组 | 15（83.3%） |
| E组 | 4 | E组 | 2（50.0%） | E组 | 4（100.0%） |
| F组 | 17 | F组 | 8（47.1%） | F组 | 10（58.8%） |

表20 别组对我组产品提出的问题清单统计

|  |  |  |  |
| --- | --- | --- | --- |
| 别组产出的问题清单数 | 别组提出的问题总数 | | |
| 4 | 0 | C组 | 0 |
| D组 | 0 |
| E组 | 0 |
| F组 | 0 |

**该阶段数据统计如表21-24所示。**

表21 本组测试评审阶段人员数据统计（1）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| E5：测试需求评审 | 测试需求评审表格设计 5.18 | | | | | |
| 姓名 | 需求变更 | 图表数 | 报告字数 | 具体工作内容 | 修订日期 | 总耗时 |
| 张雨濛 | 0 | 1 | 500 | 初步设计评审单 | 5.18 | 1 |

表22 本组测试评审阶段人员数据统计（2）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E5：测试评审 | 测试评审工作量统计 5.19-5.21 | | | | |
| 姓名 | 评审组别 | 评审意见条数 | 意见字数 | 负责范围 | 总耗时 |
| 胡俊涛 | E | 4 | 300 | 管理员测试用例 | 1 |
| 刘子渊 | F | 8 | 550 | F组全部测试用例 | 2 |
| 张雨濛 | F | 20 | 800 | F组测试需求规格说明书 | 2 |
| 叶柏威 | E | 6 | 400 | 可靠性测试用例 | 1 |
| 暴明坤 | F | 1 | 15 | F组全部测试用例 | 0.5 |
| 夏欣怡 | E | 3 | 272 | 文档逻辑与测试用例设计 | 1 |

表23 本组测试评审阶段人员数据统计（3）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| E5：测试评审 | 测试评审工作量统计 5.26-5.28 | | | | |
| 姓名 | 评审组别 | 评审意见条数 | 意见字数 | 负责范围 | 总耗时 |
| 胡俊涛 | D | 5 | 150 | D组网页功能部分测试 | 1 |
| 刘子渊 | C | 20 | 900 | C组全部测试用例 | 2 |
| 张雨濛 | C、D | 2 | 100 | C、D组测试需求规格说明书 | 2 |
| 叶柏威 | D | 14 | 600 | D组网页功能部分测试 | 2 |

表24 本组测试评审阶段人员数据统计（4）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| E5：测试评审 | 修改工作量统计 | | | | | |
| 姓名 | 修改意见 | 修改字数 | 修改代码数 | 具体工作内容 | 工作难度 | 总耗时 |
| 刘子渊 | 6 | 200 | 2 | 根据C、D、E、F评审意见修改3.8、3.9相关用例 | 一般 | 1 |
| 张雨濛 | 26 | 800 | 0 | 根据C、D、E、F评审意见修改测试需求规格说明书、测试报告及软件问题报告 | 一般 | 3 |
| 夏欣怡 | 5 | 100 | 0 | 根据C、D评审意见修改测试需求规格说明书4场景测试相关用例 | 一般 | 0.5 |

## 数据分析与说明

* + 1. 数据来源

**评审数据来源于CDEF组，表单保存在对应组别的相应目录下，由于问题反馈文档中包含问题清单中的所有内容，所以本组没有进行二次传输。**

* + 1. 数据分析

**在测试评审过程中，本组在两周的时间内对CDEF四组的测试需求规格说明书与软件程序实现进行了评审，与此同时还进行了软件设计实现的改进工作，因此时间上比较紧凑，工作量比较大。**

**基于这种情况，我们组在测试评审过程中，选择了分组并行评审的方式。在每周的评审过程中，消息聚集组与数据可视化组将分别对两组的软件进行评审，文档与示例开发组则评审两组的测试文档，然后在小组会议中交流对两组的评审结果。**

**在撰写测试文档时，我们从需求评审中吸取了经验，对文档格式规范等问题给予了更多的重视，因此评审意见总数相较于需求评审而言，得到了大幅度的降低。**

**在工作量方面，暴明坤同学在测试评审阶段主要负责和其他组之间进行协调，包括提供与维护测试环境，相对在评审方面参与较少。夏欣怡同学在复评审阶段由于特殊情况，请张雨濛同学帮忙代完成了评审工作。意见修改部分对应成员各自完成了其工作，成员工作量相差不大。**

## 问题与方法

* + 1. 如何进行评审过程，保证评审质量

**有了需求评审的经验，结合本次任务特点，我们首先对任务进行了划分，然后将每一块内容具体分配到各小组中，保证评审的完备性。各组独立进行相应模块的评审，然后最终在小组会议中进行探讨，以保证最终评审意见的质量。**

* + 1. 如何规范评审过程

**与需求评审类似，在测试评审开始之前，本组首先制定了检查单，包含四类问题：完整性、一致性、规范性以及准确性，共13条规则。在后续的测试评审过程中，小组成员都以该检查单为标准，进行评审工作。**

## 总结

**在本次实验中，通过与多组之间两轮的测试评审，大大完善了我们的测试文档，同时优化了测试文档的格式质量。通过这些测试评审，我们认识到，测试文档的编写，需要使用一套规范化的专业语言来保证测试流程的无二义性。同时也认识到，在进行测试时，需要对不确定因素进行明确，清除测试中可能出现的噪音，以免测试结果受到影响。**

# 实验六 项目计划与监控

## 概述

进度计划与控制对项目在进行阶段的任务与分配进行规划、实施、统计和调整。通过统计当前项目的进度、人员使用情况，与计划进行对比分析，判断任务分配是否合理，项目进展是否符合预期，并在出现问题时及时进行调整。

## 数据统计

由于统计任务非常多，这里只展示了一级目录情况，如图7所示。更具体的情况见项目进度project。

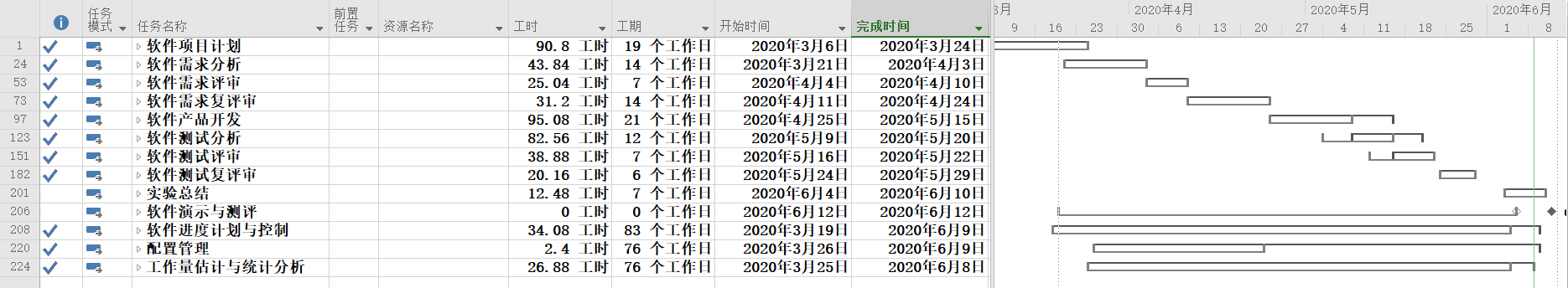


图7 项目进度Project示意图

项目计划的变更情况如表25所示。

表25 项目计划变更情况

|  |  |  |  |
| --- | --- | --- | --- |
| 项目计划版本号 | 变更情况 | 变更原因 | 时间 |
| V1.0.0 | 大致对本学期的各个阶段进行划分，并更新项目计划与需求分析阶段的部分完成情况内容 | 无 | 20190327 |
| V2.0.0 | 补全第一版中的项目计划阶段的工作，补全实验6-8的工作 | 补全上一版本未统计全的任务情况，定期更新 | 20190403 |
| V3.0.0 | 填写本周实际工作进度 | 定期更新 | 20190410 |
| V4.0.0 | 填写本周实际工作进度 | 定期更新 | 20190417 |
| V5.0.0 | 填写本周实际工作进度 | 定期更新 | 20190424 |
| V6.0.0 | 补充基线，填写本周实际工作进度 | 按老师要求对基线进行补充，定期更新 | 20190508 |
| V7.0.0 | 填写本周实际工作进度 | 定期更新 | 20190516 |
| V8.0.0 | 填写本周实际工作进度 | 定期更新 | 20190522 |
| V9.0.0 | 填写本周实际工作进度 | 定期更新 | 20190604 |
| V9.1.0 | 对软件开发阶段与测试分析阶段统计滞后的任务进行调整，使其更符合实际情况 | 按老师要求对填写不准确的任务进行重新划分 | 20190609 |

## 数据分析与说明

**本项目的进度计划管理使用MS project对项目进行计划安排和追踪，考虑到.mmp文件不易与共享查看与填写，采用飞书共享文档对任务分配与计划进行共享，并收集各成员的完成情况，最终汇总到MS project文件中，以方便组员的查看与更新。在更新方面，每周六通过共享文档对当周的任务完成情况进行统计，负责同学将其汇总更新至.mmp，并大致对下一周的任务分配进行安排。**

**本项目的进度计划目前共9个大版本，为每周一更的频率，统计当周的任务完成情况。其中在五一假期阶段停更一周。在V1.0.0版本时，根据往年以及课程安排，对本次实验的任务进行了大致规划。之后V2.0.0版本时，由于前一版本初次进行，且使用了共享文档填写的方式，组员不熟悉这种形式，使得在第一次更新时未能统计到所有任务情况，在V2.0.0版本时进行了补充。V6.0.0版本时，根据老师的建议加入了基线。V9.1.0版本时根据老师建议，将之前五一部分滞后统计导致的部分任务归类错误的情况进行了调整。其余版本为日常更新。**

**图3展示了成员的工时情况。其中上图橙色的剩余工时，为个人总结部分以及演示部分的估计工时，目前仍未完成。除此之外的工作已经完成，其中暴明坤工时91。6工时，胡俊涛79.04工时，叶柏威74.12工时，夏欣怡85.95工时，刘子渊91.04工时，张雨濛81.68工时。由于MSProject资源百分比分配的特性，统计时以0.96小时为单位。为从上图可看出，各成员之间的工时差距较小，其中暴明坤与刘子渊承担了领导的工作，花费工时也最多，其余成员各自完成了分配的任务。**

**工时燃尽图与任务燃尽图如图1所示，其中在工时燃尽图中4.27-5.11的阶段中，有一段剩余累计工时下降速度快于其他阶段的情况，这是对应的是软件开发与软件测试阶段。因为在之前的需求分析阶段，未能及时进行开发工作，以及低估了自测所需的时间，导致该段时间的工作压力较大。任务燃尽图中，有一段任务迟于基线的完成情况，是因为基线设置时未考虑到假期因素，成员在假期没有像工作日一样的工作热情，使得任务没能按预期完成。其余时间基本正常燃尽。**

## 问题与方法

**在软件开发与测试分析周，由于中间由五一假期的存在，成员存在工作热情不高的情况，包括前期的开发工作进展不如预期，导致软件开发最终在测试周的周一左右完工，测试文档的撰写最终在测试评审周的周日左右完成（本组以周六到下周五作为一个周次），这也是MS Project中两周的基线与实际完成出现了不重叠的原因。总结后认为，前期组内依赖组长进行任务分配与进度推进，在组长特殊情况暂离时，缺乏组内交流与任务推进的积极性，一定程度上导致了开发与测试阶段工作量大难以按时完成。之后刘子渊同学也辅助参与到了任务分配与推进中，并通过每晚小型会议的方式（开发与整合阶段）汇报各自进展，督促成员工作并提供交流环境，有效提高了工作效率。**

**每周定期的任务记录更新很有必要，一方面避免成员由于时间推移模糊了对自己的工作记忆，能让成员高效进行填写，另一方面能及时对当周的任务完成情况进行总结，发现存在的不足，从而能在下周开始前进行调整，使得能更好地分配下周任务。**

## 总结

项目进度管理能够很好地记录整个实验任务的分配与完成情况，并且也可为工作量统计提供数据支持。在项目最后总结时也发现，此时对于早期各个成员的所分配的任务，完成情况，实验完成中遇到的问题等记忆已经比较模糊，而定期记录的项目进度管理能够通过各种图表与数据，帮助重现实验的任务进行流程，梳理与发现工作进度中的问题，对项目的总结起到指导意义。

# 实验七 配置管理

## 概述

本项目采用GitHub进行配置管理，其中项目配置管理可具体划分为文档配置管理与代码配置管理两部分，文档配置管理由张雨濛同学负责，代码部分则是各组分别管理相应模块代码，最后统一链接到总仓库上。小组所有成员共同完成对项目配置管理的监督。

## 数据统计

项目中各阶段的制品提交情况如表26所示。

表26 各实验制品提交情况统计

|  |  |
| --- | --- |
| 实验阶段 | 提交制品数量 |
| 软件项目计划 | 软件项目计划书\*2 |
| 软件需求分析 | 需求规格说明书\*13 |
| 软件需求评审 | 需求检查单  对别组的评审结果\*4  对别组的评审反馈\*4 |
| 软件产品开发 | 设计实现方案 |
| 软件测试 | 测试需求规格说明书\*3  测试报告\*2  软件问题报告\*2 |
| 软件测试评审 | 测试文档检查单  对别组的测试需求评审结果\*4  对别组的软件评审结果\*4  对别组的评审反馈\*4 |
| 软件进度计划与管理 | 项目进度Project\*9  进度计划与控制分析报告 |
| 配置管理 | 配置管理文档\*2  配置管理总结报告 |
| 工作量估计与统计分析 | 工作量统计与分析文档\*7  工作量统计与分析总结 |

## 数据分析与说明

由于本组在文档撰写过程中采用的协作方式是所有成员在共享文档中同时更新，每名成员在完成相应模块内容撰写后填写“版本变更历史”对修订内容进行说明，与此同时文档版本号也将随之变化，而在所有成员全部完成后再由文档负责人对共享文档进行导出调整提交工作，所以可能出现文档版本数多于提交次数的情况，该情况属于正常情况在此特作说明。各文档具体版本迭代情况可参考配置管理总结报告。

## 问题与方法

在代码配置管理方面，如何使代码管理和开发模型相适配是最重要的第一个问题。主要的考虑思路是，不耦合的开发过程应该将代码放置到不同的或文件夹中，耦合程度越低，代码储存分隔应该越大。本组的消息聚集组和数据可视化组的代码和需求之间没有任何耦合，所以在开发时，为了尽可能的避免提交和评审上不必要的工作，这两组的代码分别使用两个代码仓库进行储存，使两组之间的开发解耦。之后使用git submodule将这两个代码仓库整合到主制品仓库中。

## 总结

**通过对整个项目进行分析，我们总结了以下几点：**

**首先是项目的目录结构方面，在项目初期可以参照往届的目录结构进行项目的构建，但随着项目的进行，应根据需要及时地调整目录结构以保证更好地易读性。**

**其次是代码管理方面，通过使用git submodule解耦的方法，消除了代码提交和评审上不必要的工作。同时也允许两个工作组使用稍有不同的开发风格和不同的开发步调。整体上提升了全组的生产力。**

# 实验八 统计分析

## 概述

**根据项目内容与互评，我们完成了实验一到实验八的各项内容。实验一为项目设计与需求分析阶段，进行了项目计划与需求分析；实验二为需求评审阶段，进行了软件需求的互评审。实验三为软件产品开发阶段，完成了项目的具体开发；实验四为测试需求分析阶段，对软件进行了测试设计、实现与文档撰写；实验五为测试评审阶段，进行了软件测试的互评审；实验六至八贯穿于整个实验过程。**

**组内按照个人任务，分为消息聚集组、数据可视化组与文档示例与开发组，每组两名成员。其中消息聚集组成员为暴明坤与刘子渊，数据可视化组成员为胡俊涛与叶柏威，文档与示例开发组成员为夏欣怡与张雨濛。消息聚集组负责飞书相关节点开发，数据可视化组负责图表类节点开发，文档与示例开发组负责文档模板、汇总以及实验6-8相关内容。其余工作如评审方面，全部成员共同参与。工作量统计中以个人为单位进行统计。**

## 数据统计

工作量统计结果如表1所示。在总结文档中只展示了汇总表，分表详见工作量统计与分析汇总文件。

## 数据分析与说明

工作量统计共计设计了17个表格，对各个阶段任务进行统计，并针对六个方面，对统计内容进行整合分析。在统计方式上，采用了共享文档的方式，更新统计表格并提醒全体成员填写的方式进行数据收集，每一阶段进行校验、导出与更新。总体而言，各成员在各个方面参与程度有着一定的差别，但都完成了自己该做的工作，没有出现工作量远超或者明显少于其他组员的情况。其中暴明坤与刘子渊在开发与小组领导上做了更多工作，在最终统计上优于其他成员，其余成员相对平均。

## 问题与方法

**在表格设计最初，考虑到任务难度本身存在差异，而组员各自对自己分到的任务难度有更直观的体会，故设置了工作难度项，希望组员对自己的任务难度进行评估。但实际填写中发现，各组员较为谦逊，即使分配到的任务各不相同，但在难度评估项中的填写非常一致，导致该项统计没有实际意义。但如果交由单人设定，由于对各成员实际面对、解决的问题并不了解，容易出现偏颇。如何对这种没有定量标准的数据进行正确统计，没有找到很好的方法，目前采用数据与工时两个方面，对工作难度导致的数据少但实际耗时长的任务进行工作量评估。**

**提前设计统计表格，根据任务实际情况进行调整。提前设计统计表格，即使不够完善，但能够起到一个参考价值，在当周实际进行任务的时候能实时发现表格设计存在的缺陷并及时进行调整。如果在当周结束时才去设计表格，没考虑周全的可能性会更大，而且此时成员已经开始填写统计表格。当成员填写完后才发现统计上的缺陷，会额外增加工作量。**

**按时收集数据。及时提醒成员在当周结束时进行填写，并及时进行校对。当周刚刚完成工作时，对自己做过什么，以及相对应的文档代码位置有清晰的记忆，这时填写统计事半功倍，更加准确，也方便成员间进行互相监督。**

## 总结

工作量统计与分析是软件工程中重要的一部分，设计全面，统计及时的工作量表格能够记录成员在工作中的参与情况，反应真实的成员贡献度，对下一步的任务分配起到指导意义。同时，也能促进组员间与组员自身的激励，提高组员的工作积极性。在总结阶段，也能有据可依，对成员的贡献度做出正确评估。

# 课程总结

本门课的课程思路清晰，教学目的明确。通过模拟的形式，我们了解了软件开发的流程、涉及到规范以及可能遇到的问题及其处理方法。除此以外，通过这种形式的实验，我们的确可以在一定程度上对软件开发有一定的总体了解，并且可以锻炼一系列的相关能力，例如沟通能力、定义能力和协调组织能力等。