**S18小组项目测试文档汇总**

编写 崔世昂 2020 年 05 月 27 日

审核 年 月 日

批准 年 月 日

版本控制

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 日期 | AMD | 修订者 | 说明 |
| 1.0 | 2020/5/27 | A | 崔世昂 | 添加项目测试文档汇总 |
| 1.1 | 2020/5/31 | M | 崔世昂 | 添加个人总结、两项声明等内容 |

注：（A-添加，M-修改，D-删除）

目录

[S18小组项目测试文档汇总 1](#_Toc16207)

[版本控制 2](#_Toc21337)

[目录 3](#_Toc10269)

[1 独创性声明 4](#_Toc17822)

[2 使用权声明 4](#_Toc28257)

[3 测试计划&测试方案 5](#_Toc12851)

[4 缺陷报告&质量报告 5](#_Toc9890)

[5 配置管理 5](#_Toc23316)

[6 会议纪要 6](#_Toc21553)

[7 团队总结 7](#_Toc625)

[8 个人总结 7](#_Toc21528)

1 独创性声明



2 使用权声明



3 测试计划&测试方案

      
（注：兼容性测试计划 嵌入在 软件测试方案设计 内）

4 缺陷报告&质量报告

    

5 配置管理

我在项目初期就严格地使用Git进行配置管理，虽然往往是我一个人在push和fetch、merge，但我的初衷是大家一起pull request的。  
github仓库地址：  
<https://github.com/CORIGIN-And-His-Little-Friends-2020-4-8/SQ-ETteam2020COST.git>  
在仓库中可以看到所有的分支的commit、merge等记录

6 会议纪要

我们并没有采用会议软件进行严格的会议，但我使用TIM建立了一个讨论组 2020软件质量与评测技术小组。任务分发以及通知我都是在群里进行说明的，大家也能有效地进行反馈。我觉得虽然使用会议软件能更加有效的面对面交流，但是其时间分配并不像群聊一样灵活。我最初的想法是完全采用分布式管理，也就是我将任务分配给组员，给他们一个DDL，然后让他们采用Git的fork、branch、pull request机制自由、独立地支配自己的时间并向我发出提交请求。我充当了一个配置管理员的角色（只有我能直接write仓库，其他人只能pull request），我会将他们的成果整合进他们的成果分支，然后阶段性地（比如中期报告时）将他们的成果merge到master分支。当然这只是我的初衷，实际操作也差不多，只不过是他们的pull request变成了我的Push，大家还是觉得直接TIM传文件给我更方便。  
其实这一块根据我在Github上Projects分支的各项计划commit记录就能理得清清楚楚。  
2020/4/20 制定项目测试总计划，分配测试子计划任务

2020/4/27 按测试子计划分配测试用例设计和测试报告撰写任务

2020/5/6 中期成果验收，准备提交中期报告

2020/5/27 准备收尾工作

7 团队总结



8 个人总结

     