**迭代计划（简化版）**

制定日期：2020-11-17

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 14 | 项目名称 | PC Logo |
| 迭代名称 | 构造迭代 | 计划起止日期 | 2020.11.18-2021.01.04 |
| 任务、进度安排和人员分配：   |  |  |  |  | | --- | --- | --- | --- | | No | 任务 | 起止日期 | 人员 | | 1 | 小组讨论分工，编写迭代计划 | 2020.11.18 | 全体成员 | | 2 | 完善命令行操作各种指令的实现 | 2020.11.19-2020.11.20 | 吴天龙 | | 3 | 实现双人网络联机（仅包括连接，可不实现画图） | 2020.11.19-2020.11.20 | 吴泽川 | | 4 | 学习Qt单元测试工具QTest | 2020.11.19-2020.11.20 | 苏浩然 | | 5 | 对调试功能进行尝试实现 | 2020.11.19-2020.11.20 | 张哲溢 | | 6 | 对界面进行进一步美化设计 | 2020.11.19-2020.11.20 | 吴泽一 | | 7 | 实现简单的命令文件操作 | 2020.11.21-2020.11.22 | 吴天龙 | | 8 | 完善网络连接功能 | 2020.11.21-2020.11.22 | 吴泽川 | | 9 | 初步对单元测试进行应用 | 2020.11.21-2020.11.22 | 苏浩然 | | 10 | 实现单步运行功能 | 2020.11.21-2020.11.22 | 张哲溢 | | 11 | 继续对界面进行设计、美化 | 2020.11.21-2020.11.22 | 吴泽一 | | 12 | 对最终界面进行评审、改进 | 2020.11.23 | 全体成员 | | 13 | 完善命令文件功能 | 2020.11.24-2020.11.25 | 吴天龙 | | 14 | 实现双人联机单海龟绘图 | 2020.11.24-2020.11.25 | 吴泽川 | | 15 | 制作Qt单元测试demo | 2020.11.24-2020.11.25 | 苏浩然 | | 16 | 初步实现断点功能 | 2020.11.24-2020.11.25 | 张哲溢 | | 17 | 撰写帮助文档 | 2020.11.24-2020.11.25 | 吴泽一 | | 18 | 对命令行绘图、命令文件绘图进行完善改进 | 2020.11.26-2020.11.27 | 吴天龙 | | 19 | 进一步完善双人联机单海龟绘图 | 2020.11.26-2020.11.27 | 吴泽川 | | 20 | 对已实现的功能进行单元测试 | 2020.11.26-2020.11.27 | 苏浩然 | | 21 | 完善调试功能 | 2020.11.26-2020.11.27 | 张哲溢 | | 22 | 对尚未设计、实现的用例进行设计，并绘制相关设计模型 | 2020.11.28-2020.12.04 | 苏浩然 | | 23 | 根据设计模型，实现联机相关所有功能，并进行单元测试 | 2020.11.28-2020.12.04 | 吴泽川 | | 24 | 根据设计模型，实现命令行、命令文件绘图的所有功能，并进行单元测试 | 2020.11.28-2020.12.04 | 吴天龙 | | 25 | 根据设计模型，实现文件调试的基本功能，并进行单元测试 | 2020.11.28-2020.12.04 | 张哲溢 | | 26 | 根据设计模型，实现语音输入绘图的所有功能，并进行单元测试 | 2020.11.28-2020.12.04 | 吴泽一 | | 27 | 对当前成果、进度进行小组讨论、评估，对不足之处进行改进 | 2020.12.05-2020.12.11 | 全体成员 | | 28 | 根据设计模型，实现子过程相关功能，并实现对子过程的调试，并进行单元测试 | 2020.12.12-2020.12.18 | 张哲溢、吴泽川、苏浩然 | | 29 | 实现帮助、指南等小功能，并进行单元测试 | 2020.12.12-2020.12.18 | 吴天龙、吴泽一 | | 30 | 对当前成果、进度、测试等进行组内评估，改进完善 | 2020.12.19-2020.12.25 | 全体成员 | | 31 | 各成员共同撰写单元测试报告 | 2020.12.19-2020.12.25 | 全体成员 | | 32 | 实现作品集等创新功能，并进行相关测试 | 2020.12.19-2020.12.25 | 吴天龙、吴泽一 | | 33 | 编写系统测试用例、进行系统测试 | 2020.12.19-2020.12.25 | 苏浩然、吴泽川、张哲溢 | | 34 | 撰写系统测试报告 | 2020.12.19-2020.12.25 | 苏浩然、吴泽川、张哲溢 | | 35 | 对测试结果、报告等进行组内评估、改进 | 2020.12.26-2020.12.29 | 全体成员 | | 36 | 对迭代所有成果进行评估，撰写迭代评估报告 | 2020.12.30-2020.12.31 | 全体成员 | | 37 | 制作验收答辩PPT，撰写项目总结报告，录制演示视频等 | 2021.01.01-2020.01-04 | 全体成员 | | | | |
| 预期成果：  文档与模型：《Vision文档》、use-case模型、《软件需求规约》、《软件架构文档》、编程规范、UML分析与设计模型、《单元测试报告》、《系统测试用例》、《系统测试报告》  代码：软件代码（包括源代码和可执行程序）、单元测试及系统测试相关代码  迭代：《迭代计划》、《迭代评估报告》  项目：《项目总结报告》、验收答辩PPT、预录系统演示视频等 | | | |
| 主要的风险和应对方案：   1. 进度风险。项目有可能因为时间问题难以在截止日期前完成迭代要求。缓解应急措施：加强团队沟通与交流，及时召开小组会议，及时制订迭代计划，估算团队开发速率和项目进度，及时修正迭代计划，严格按照迭代计划执行。当进度风险十分急迫时，可以采取增加工作时长的方法来应急。此外，可以视情况放弃优先级较低的用例实现。 2. 质量风险。项目有可能无法达成某项质量指标要求。缓解应急措施：严格执行符合要求的软件开发流程，认真进行code review，及时对产品代码进行评审，进行全方面的严格的测试，及时进行单元测试，确保单元测试的高覆盖率，及时与需求方沟通并确认质量要求。 3. 技术风险。小组成员有可能对相关技术了解不够充分，导致软件开发出现困难。缓解应急措施：提前学习相关技术，对团队成员的技术能力进行充分的预判，在使用不熟悉的技术前进行充分的学习、调试。在遇到难以自己克服的困难时，及时向助教、老师或者其他小组成员寻求帮助。必要时，可以寻找某些技术的替代方案。 4. 需求风险。小组成员有可能在工作中，对需求没有理解充分，需求有可能因为种种原因在项目开发过程中发生变更。缓解应急措施：及时地召开小组会议对产品需求进行评估，认真评估项目相关的需求文档，加强小组成员之间的沟通，对项目需求充分理解。采用合理的软件开发流程，采用演化模型等相关方式，以更好地应对需求变更。 | | | |