**迭代计划**

　　　　　　　　　　　　　 制定日期：2022.11.7

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 6 | 项目名称 | 交大林檎——疫情互助平台 |
| 迭代名称 | 技术原型迭代 | 计划起止日期 | 2022.11.7 ~ 2022.12.13 |
| 预期成果：  软件架构文档一份  迭代评估报告一份  前、后端源代码  可直接部署测试的服务端程序 | | | |
| 任务、进度安排和人员分配：   |  |  |  |  | | --- | --- | --- | --- | | No | 任务 | 起止日期 | 人员 | | 1 | 选择项目使用的语言、框架和其他工具 | 11.7 | 全体成员 | | 2 | 调研软件架构，确认各模块接口；对技术难点给予评估，并给出初步的解决方案。 | 11.8—11.9 | 全体成员 | | 3 | 确认人员详细分工、编码规范。剩余时间留作机动，解决如技术难点等可能存在的遗留问题 | 11.10—11.11 | 全体成员 | | 4 | 撰写、评审软件架构文档 | 11.12 | 撰写人：敖晟昊  评审：全体成员 | | 5 | （前端）实现主页 | 11.13—11.15 | 黄川懿，孙济宸 | | 6 | （前端）实现个人需求发布页 | 11.15—11.17 | 孙济宸 | | 7 | （前端）实现登录、注册页 | 11.17—11.21 | 黄川懿 | | 8 | （前端）实现需求发布页 | 11.18—11.19 | 孙济宸 | | 9 | （前端）实现提供管理页 | 11.19—11.21 | 孙济宸 | | 10 | （前端）实现个人信息页 | 11.21—11.23 | 黄川懿 | | 11 | 前后端联调 | 11.23—11.27 | 全体成员 | | 12 | （后端）物品信息建模 | 11.13—11.15 | 敖晟昊 | | 13 | （后端）实现多级物品类别 | 11.15—11.17 | 敖晟昊 | | 14 | （后端）实现物品信息管理 | 11.17—11.23 | 敖晟昊 | | 15 | （后端）实现注册功能 | 11.13—11.17 | 王俊智 | | 16 | （后端）实现登录功能 | 11.17—11.20 | 王俊智 | | 17 | （后端）实现密码修改、密码忘记功能 | 11.20—11.23 | 王俊智 | | 18 | 前后端联调 | 11.23—11.27 | 全体成员 | | 19 | 物品信息发布的测试和修改 | 11.27—12.1 | 敖晟昊，孙济宸 | | 20 | 登录和注册的测试和修改 | 11.27—12.1 | 王俊智，黄川懿 | | 21 | 整体软件功能测试 | 12.1—12.5 | 全体成员 | | 22 | 撰写迭代评估报告 | 12.5—12.7 | 全体成员 | | 23 | 修改软件架构文档 | 12.7—12.9 | 敖晟昊 | | 24 | 准备验收材料、答辩PPT | 12.9—12.10 | 敖晟昊 | | 25 | 确认材料、提出修改意见并修改 | 12.10—12.12 | 全体成员 | | | | |
| 主要的风险和应对方案：   1. 技术难点解决方案无法快速给出   ——在评估阶段做出基本的难点明确，在后续学习、具体实现中结合实践考虑解决方案；或是绕过一开始提出的难点做相同、相似功能的其他实现。   1. 技术原型实现中遇到未预期到的困难   ——在项目设计中做需求分级，减少基本需求遇到这类问题的概率；微调项目结构，使得在功能基本不发生改变时可以回避困难。   1. 模块接口无法完全明确并完全地指明   —— 使用抽象层级较高的描述做架构设计；使用面向接口编程的设计方案，在架构设计中考虑简单、必要的接口，以对具体功能函数做一层封装；在代码实现中做定期迭代，按需要调整模块间接口关系。 | | | |