|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编辑记录 | | | | | |
| 序号 | 修改原因 | 版本 | 日期 | 编辑成员 | 备注 |
| 1 | 创建 | V1.0 | 10.12 | 江玙璠 | 新建本文档，初步制定计划 |
| 2 | 更新 | V1.1 | 10.18 | 江玙璠 | 第一次滚动更新计划 |
| 3 | 更新 | V1.2 | 10.25 | 江玙璠 | 第二次滚动更新计划 |
|  |  |  |  |  |  |

**迭代计划（简化版）**

　　　　　　　　　　　　　制定日期：10.12

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 10 | 项目名称 | 一心·时间管理APP |
| 迭代名称 | 技术原型迭代 | 计划起止日期 | 10.11-11.16 |
| 任务、进度安排和人员分配：   |  |  |  |  | | --- | --- | --- | --- | | No | 任务 | 起止日期 | 人员 | | 1 | 初步制定技术原型迭代计划 | 10.12-10.13 | 江玙璠 | | 2 | 概念建模 | 10.14-10.16 | 陈嘉轩、沈佳威 | | 3 | 进一步完善界面原型 | 10.14-10.17 | 朱益成 | | 4 | 评审概念模型 | 10.18 | 朱益成、江玙璠 | | 5 | 更新迭代计划 | 10.18 | 江玙璠 | | 6 | 完善修改概念模型 | 10.19 | 江玙璠 | | 7 | 用例分析建模——分析类识别 | 10.19 | 陈嘉轩 | | 8 | 用例分析建模——时序图与通信图 | 10.20-10.21 | 朱益成、江玙璠、沈佳威 | | 9 | 用例分析建模——VOPC类图 | 10.22 | 陈嘉轩 | | 10 | 完善界面细节部分，调整组件 | 10.23-10.25 | 江玙璠、朱益成 | | 11 | 学习使用Flutter与安卓原生代码交互 | 10.24-10.25 | 沈佳威 | | 11 | 更新迭代计划 | 10.25 | 江玙璠 | | 12 | 选择架构风格、设计架构视图 | 10.26-10.27 | 全体成员 | | 13 | 选定或撰写编程规范 | 10.27-10.28 | 全体成员 | | 14 | 设计、选择关键算法 | 10.26-10.31 | 朱益成 | | 15 | 编写软件架构文档 | 10.28-10.29 | 陈嘉轩 | | 16 | 评审软件架构文档 | 10.30-10.31 | 全体成员 | | 17 | 更新迭代计划 | 11.1 | 江玙璠 | | 18 | 搭建软件架构——服务器端部分 | 11.2-11.5 | 沈佳威、陈嘉轩 | | 19 | 搭建软件架构——客户端部分 | 11.2-11.5 | 江玙璠、朱益成 | | 20 | 搭建软件架构——前后端连接 | 11.5-11.7 | 沈佳威、朱益成 | | 21 | 实现算法（典型用例） | 11.2-11.7 | 陈嘉轩、江玙璠 | | 22 | 编写测试用例、测试技术原型 | 11.2-11.7 | 沈佳威、朱益成 | | 23 | 更新迭代计划 | 11.8 | 江玙璠 | | 24 | 编写迭代评估报告 | 11.7-11.13 | 全体成员 | | | | |
| 预期成果：  概念模型  架构视图  软件架构文档  技术原型  《迭代评估报告》  《迭代计划》 | | | |
| 主要的风险和应对方案：  **1. 技术风险**：要实现设计中的大部分功能，最关键的是涉及到手机权限的获取，包括摄像权限、录音权限、位置权限等等，能否合理获取这些权限关系到整个项目最终能不能符合预期地实现。  **缓解应急措施：**后续开发、测试的时候需要使用实机的开发者模式。当最终需要获取权限时，应当通知用户分配权限。涉及部分很难获取的权限的功能需要想出可行的替代措施。例如将控制锁屏改为锁屏通知等方法。  **2. 人员风险：**小组成员大部分都各自有选修课，能够集中在一起面对面交流、开发的时间比较有限，线上交流的效率又比较低。此外其他课程的任务也会影响成员的工作效率和时间安排。  **缓解应急措施：**尽量做到合理的任务规划，在有限的可以聚集的时间内重点集中解决一些需要集体讨论、集体工作的问题，让小组成员在独自开发的时间都有合理的任务安排。必要时利用线上会议进行讨论、协同开发。  **3. 时间风险：**目前还不能完全预估一些任务所需的时间，且项目开发的时间安排也受到其他课程课业安排的影响，可能存在无法按时完成预期的情况。  **缓解应急措施：**尽量细化近期开发计划，频繁地根据实际情况、开发进度调整迭代计划。在计划中增加对迭代计划的定期更新，以期能够灵活地实施迭代计划，提高对时间的利用率。  **4. 需求风险：**项目目前的特色不够突出，更像是一个现有产品的简单整合。可能存在未识别的用户需求，存在需求变更的风险。  **缓解应急措施：**在正式开展技术原型的开发前，再次对上一次迭代的结果进行反思讨论，同时再次深入分析竞品，通过小组讨论、头脑风暴等方式，挖掘更多的潜在需求，明确项目的特色。 | | | |