**迭代评估报告（简化版）**

　　　　　　　　　　　　　　　　　　评估日期：10.9

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 10 | 项目名称 | 一心·时间管理APP |
| 迭代名称 | 界面原型迭代 | 实际起止日期 | 9.16-10.8 |
| 任务达成情况：（完成的任务、实现的功能、进度、质量等）   1. 通过讨论确定项目的基本信息，并设计发放了问卷进行了市场调研，同时依据反馈的数据明确产品的需求、定位，完善产品的功能设计，编写了调查问卷报告。 2. 选定了项目所需的语言（Dart）、工具以及框架（Flutter），并且通过小组共同学习比较高效地完成了对陌生技术的熟悉和使用。 3. 编写了《Vision文档》以及《软件需求规约》，并在助教的建议下进一步完善。 4. 依据《Vision文档》，利用PowerDesigner建立了清晰直观的Use-case模型。 5. 完成了界面的设计和界面原型的开发，实现了登录界面、闹钟列表、闹钟设置、“远离手机”功能界面以及个人界面，比较成功地完成了预期的目标。 | | | |
| 评审/测试的结果：（执行了哪些评审和测试？评审和测试的结果如何？）  暂无 | | | |
| 问题、变更和返工：（遇到的问题、发生的变更、是否需要返工等）  问题：   1. 编写《Vision文档》和《迭代计划》，对于我们小组的所有成员来说都是第一次，在编写时尽管有模板供作参考，但是仍有一些内容写得比较模糊、简略。特别是人员安排和优先级风险指定的部分比较粗略，不够明确详尽；对竞品的分析也不够全面。 2. 《迭代计划》中，对一些安排的任务量的估计出现了偏差，文档更新也不够及时。虽然总体上没有超出第一次迭代的时限，但部分任务并没有完全严格按照计划指定的时间安排完成。 3. Flutter开发的经验不足。在使用开发者社区开源的一些组件时，由于组件本身的复杂性，以及满足适配本项目的需求，不得不引入更多的组件，进行一些个性化的改动，使得复杂程度大为增加。   变更：   1. 完善修改《迭代计划》，根据实际实施情况，更为具体合理地指定了任务的成员安排，细化了任务的时间安排，改进了风险的识别和优先级。 2. 完善《Vision文档》，增加了对竞品——“火箭闹钟”的分析和比较，细化了优先级设置，添加了对系统可用性的详细描述。 3. 依照任务的实际完成情况和对剩余的任务量的进一步分析评估，重新编写了《迭代计划》的任务安排表。 4. 暂时弃用复杂的组件，在满足项目需求的前提下，先使用简单的组件构建界面原型并积累Flutter开发经验，以期在后续开发过程中逐渐变更丰富，进一步调整、完善界面原型。   返工：  暂无 | | | |
| 经验和教训：   1. 在迭代的计划上，应该经常对迭代计划的任务安排进行重新评估，结合项目实际的开发进度和需求，调整、细化时间安排，让计划能够更好地指导开发工作的进行。 2. 在使用复杂的组件前，可以先单独做一个demo，预览一下实现效果同时熟悉使用方法，评估一下开发的复杂性与实用性，然后再决定是否实际应用到项目中。在开发前期或者时间紧张的情况下，合理地使用简单的组件进行替换。 3. 编写迭代计划及其他文档时，考虑要尽量详细完善，不能因为自己是开发小组的成员，对项目比较熟悉，就在一些需要明确指定的地方简略言之，导致给项目的迭代引入风险。 | | | |
|  | | | |