<事务管理清单>

项目总结

版本 <1.0>

**项目总结**

1. **说明**

此文档主要内容是在此次开发过程中遇到的问题总结，主要有团队合作问题和技术问题，也包括其他种类问题，下文中各种问题排列顺序为开发过程中出现该问题的大体时间。

**二、问题及解决**

（1）**团队合作问题：**由于我们小组是临时拼凑的小组，组员之间也不是很熟悉，所以在项目的最初确定需求方向时遇到了麻烦。

**解决：**在6月24日上午的集合时间内，我们第一次集合到了一起，在不同的需求方案中最终通过讨论各种利弊，并最终投票的方式确定需求方向。在组员彼此间不熟悉的情况下，投票的方式很 是有效。

1. **技术问题**：小组组员全是第一次接触微信小程序，也没有参加之前的微信小程序大赛，所以对小程序的相关知识等需要现学，尤其是有关后端开发的JavaScript。

解决：在项目开始前，我们便从零开始学习小程序开发，通过网上一些教学视频和微信小程序官方文档等进行了学习。

1. **团队合作问题：**在第一次迭代，虽然确定了设计方向，但由于分工不具体导致了几乎没有进度，在第一次迭代即将结束时，我们才勉强拼凑出需提交的材料。

**解决：**实际上分工不具体的根本原因还是需求定义不具体，所以我们进一步完善需求定义，并以此为基础进行了团队分工，相比之前大家做重了或是有些部分没有顾及的情况改善了不少。

1. **团队合作问题：**项目进程管理问题，我们对github的使用不熟练，没有很好利用其功能。

**解决：**我们一位组员专门研究了github，并作为github组里的master一步步指导我们上传代码文档等，且会在github上将分工、本期目标、需求等各种信息显示出来，帮我们了解各自的待办事项等，熟练应用github后，它带给我们的帮助着实显著。

（5）**技术问题：**“日历”功能是我们所做小程序的一个重点内容，很多其他功能都在其基础上实现的，所以实现日历功能比想象中麻烦得多。

**解决：**我们查询了创作日历程序应有的过程，通过微信小程序官方设计文档的帮助一步步了解日历功能的实现；并在写日历功能的时候，由于实在很复杂，所以我们对代码的注释很看重，将所有重要地方的注释都写得很细，方便在之后为日历界面添加其他功能时一目了然代码内容。

（6）**技术问题：**前端后端的整合，由于在界面上分工，我们分为前端界面和后端，所以在程序完成运行，也就是整合时，对接出现问题。

**解决：**最好的办法一个是多沟通在微信群里直接沟通，所有人都能看见，更容易帮助发现问题；还一个就是代码的注释一定清晰细致。

1. **技术问题：**临近开发截止时间，我们设计的小程序功能不够丰富。

**解决：**在发现问题后，小组开了会，详细讨论了如何进一步充实需求，重拟需求定义，将小程序功能完善的情况下尝试额外添加了新的功能。

1. **文档问题：**文档内容没有符合要求，内容较少。

**解决：**按照提供的文档模板，将文档内容完善充实。

1. **技术问题：**最主要的问题还是代码实现的问题，小程序开发到最后，添加很多辅助功能时，最困难的还是代码的编写。

**解决：**上网搜索有关功能所需要的函数，一点点稳定地从头做起，逐步实现各个功能。

**三、总结**

此次小学期我们经历了一次较为完整的软件开发进程，从需求、计划到代码测试及总结，更切实地体会到了软件工程的定义：它是去研究和应用如何以系统性的、规范化的、可定量的过程化方法去开发和维护软件的过程。因为只有以这样的态度，才能在经历过软件危机的现在研发出一款实用的软件。

这次的实践让我们对软件工程导论这门课程甚至软工专业都有了更深的认识，在第一次迭代中，我们完成得很差，主要是我们缺乏对迭代的定义的认识。在我们原先的认识中，将迭代理解为该项目的几个步骤，但这是我们对它的误解。每一次的迭代都应该包含对该项目的需求分析、软件开发、测试。有了这样的认识，我们在接下来的两次迭代中，都挖掘了需求，也开发了许多的功能。

至于在此次开发过程中遇到的种种问题，像是对之后进行软件开发任务的警戒，在之后遇到同样问题的时候，凭借经验便可妥善处理，问题和解决积累的越多，以后面对问题束手无策的可能便会越小。