**迭代评估报告**

　　　　　　　　　　　　　　　　　　评估日期：2020/11/16

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 20 | 项目名称 | 时间管理APP |
| 迭代名称 | 技术原型迭代 | 实际起止日期 | 10/16~11/16 |
| 任务达成情况：（完成的任务、实现的功能、进度、质量等）   1. 完成了核心需求的实现 2. 对部分中等优先级需求选择性部分实现 3. 部分中等优先级需求还没有完全实现 4. 对部分低优先级需求搭建了雏形 5. 完成了编程规范的讨论与完善 6. 撰写了软件架构文档 7. 撰写了UML类图设计 8. 撰写了用例分析建模 | | | |
| 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？）   1. 对产品成果分三次进行了评审；成果主要为对已有功能找出漏洞和不足，为下阶段开发提出方向 2. 对软件编程规范进行了评审；成果主要为对已有杂乱的软件文件结构和命名等进行了重构，选择了共同遵守的编码开发规范 3. 对软件架构设计进行了评审；成果为优化了架构方案的选择，主要为客户端几种常用框架，服务端部署模式等做出了选择 | | | |
| 问题、变更和返工：（遇到的问题、发生的变更、是否需要返工等）  遇到的问题：   1. 项目进度管理不佳：成员没能每周按时完成迭代计划内容，导致迭代计划不能如期执行，被迫修改迭代计划的安排，导致后期迭代计划较为繁重，被迫暂缓了部分中等优先级任务的实现，一定程度上影响了软件质量 2. 涉及多人业务交互部分沟通不及时，没有对接好，导致多次修改代码以及重复代码和返工 3. 架构、技术问题解决速度较慢，效率较低，一定程度上影响了进度和质量，有时甚至需要更换技术选型 4. 制定编码规范前文件结构管理混乱，后来经过重新组织才较为合理 | | | |
| 经验和教训：   1. 以后要注重进度管理，指定更细粒度的更符合实际的迭代计划，同时全体成员相互督促按照迭代计划进度完成任务。 2. 成员间要加强沟通，尤其是涉及到业务交叉部分，一定要及时对接相关负责人，协调有关业务逻辑的设计和完善 3. 技术架构做好前期评估和调研，遇到无法独立解决的技术问题多和有经验的老师同学交流沟通，避免闭门造车。 | | | |