**迭代评估报告（简化版）**

　　　　　　　　　　　　　　　　　　评估日期：

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 13 | 项目名称 | MY PC Logo |
| 迭代名称 | 界面原型迭代 | 实际起止日期 | 2020/9/15-2020/10/5 |
| 任务达成情况：（完成的任务、实现的功能、进度、质量等）   1. 调研、分析、定义需求任务完成，编写《调研结果（软件需求调研）》 2. 语言、工具和框架选择：c++、c#，python，平台工具不限，.Net框架 3. 界面设计，实现界面原型，见UI-img文件夹 4. 完成《Vision文档》、use-case模型、《软件需求规约》和《迭代评估报告》 | | | |
| 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？）  小组对《Vision文档》、use-case模型、《软件需求规约》和《迭代评估报告》等文档进行评审  根据文档，项目开发过程中的难点：   1. 客户端下载以及版本更新 2. 高并发的解决 3. 进阶需求语音识别需要外部设备（麦克风）接入 4. 用户合作交互设计保持一致性 | | | |
| 问题、变更和返工：（遇到的问题、发生的变更、是否需要返工等）   1. 开发语言c++不支持语音识别，添加python语言，不需要返工 2. 后续开发代码风格不统一，建立《词汇表》，《项目API规范》， | | | |
| 经验和教训：  1.之前做项目的时候没有提前写文档的习惯，本次迭代在开始项目之前先进行了文档撰写，是一种全新的编程思路，因此在迭代一开始我们遇到了很多问题，如不知道很多后续功能的实现架构等。但是经过文档的完善，我们对项目的整体架构有了远见，减少了后期的返工或重做，我们决定以后按照这种模式进行项目设计，以提高编程效率。  2.在本次迭代中小组也暴露了缺点，如组员之间磨合适应，缺乏交流，小组合作效率并不高，我们决定为了以后提高效率每周至少租两天晚上的小组自习室，进行小组进度、技术实现的讨论以及完成任务的报告。 | | | |