**迭代评估报告**

　　　　　　　　　　　　　　　　　　评估日期：2020年10月9日星期五

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 第五组 | 项目名称 | 云作业平台 |
| 迭代名称 | 迭代一：界面原型迭代 | 实际起止日期 | 9月21日 – 10月9日 |
| 任务达成情况：  完成的任务：  详细制定本次迭代的计划与分工  学习use-case模型并制作本项目的use-case模型  编写vision文档  编写软件需求规约  选择项目开发所需要的语言，工具，框架  分析项目的功能点并划分，设计页面  完成实现界面原型的开发  改进界面原型  改进软件需求规约  综合助教的意见改进vision文档  编写迭代评估报告  制作迭代成果展示ppt  实现的功能：  本次迭代以界面原型的设计与实现为主，暂时不涉及功能领域  进度：  完成了本项目所需要的web前端界面原型的开发设计  完成了use-case模型的建立  完成了vision文档与软件需求规约  完成了迭代计划中所安排的所有工作，开发进度符合预期  质量：  在规定的时间内较高质量的完成了所有计划中安排的任务  界面原型功能点覆盖完整，设计较为美观，完成质量较高 | | | |
| 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？）  评审/测试：  小组内部对整合的界面原型进行了简单的评审  结果：  界面原型的开发成果符合预期，效果较好 | | | |
| 问题、变更和返工：  1. 在写软件需求规约时发现有部分内容并不能在项目进行的开始就完成，需要进行更加细致的讨论与研究后才可以完成。变更为对迭代计划进行了一些修改，添加了改进软件需求规约这一项。不需要返工。  2. 在进行界面原型开发时，组件库material UI的配置出现问题。为了避免进度风险，及时改为使用antd组件库，不需要返工。  3. 前端展示用的静态页面中静态数据因开发人员不同而有所出入，不影响界面原型的展示，但不同页面的数据不同步。由于在改为动态数据该问题不再存在，不需要变更与返工。  4. 前端框架的选择，在前端主流框架react与vue中选择，由于小组成员更熟悉react而最终选择react。 | | | |
| 经验和教训：  1. 在之后制定迭代计划时，要对安排的各项内容有更深入的认知与了解，明确各个任务的前置关系，避免因为安排不当无法完成当前任务而打乱原先的计划。  2. 在遇到环境配置问题时，若无法解决，及时考虑更换技术栈以避免进度风险。  3. 多人开发时对于共享的数据，接口等信息要及时协商好，避免因数据，接口等信息不同步而被迫返工，延迟开发进度。  4. 在前后端开发时，要事先定好接口的数据类型，对照接口进行代码的开发，避免因前后端数据不匹配而引发的各种问题。  5. 在遇到选择时要综合成员的熟悉程度与使用人数（查找相关资料的难易程度）进行考虑，抉择。 | | | |