**迭代评估报告（简化版）**

　　　　　　　　　　　　　　　　　　评估日期：2020/10/9

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 第三组 | 项目名称 | 小箱交--云作业平台 |
| 迭代名称 | 第一次迭代“界面原型迭代” | 实际起止日期 | 2020.9.7-2020.10.9 |
| 任务达成情况：（完成的任务、实现的功能、进度、质量等）  1. 确定了本次迭代的迭代计划，将完成各项任务的时间粒度细分到2~3天，确定了各项任务的负责人。  2. 确定了本项目的功能需求，完成了use-case模型的设计；在进度范围内完成了一版迭代，但之后由于发现设计不完全，因此在预期时间之外又重绘use-case，总体上进度跟进良好；完成质量正常。  3. 完成了《vision》文档，完成对产品的前景展望，对产品功能和质量作出规划。9.16开始学习如何书写前景文档，9.23完成并进行组内评审，9.24进行修改。课上老师点评后又进行了修改，到10.2完成修改。  4. 完成了《软件需求规约》，完成了关于本项目的需求功能的详述以及系统性能参数的说明。从9月28日开始，到10月4日完成《软件需求规约》，之后开始组内评审，并于10月9日完成了组内评审。  5. 完成了界面原型设计，包括登录界面，注册界面，控制面板界面，提交作业界面，批改作业界面，发布作业界面等。 | | | |
| 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. 在项目需求的确定以及use-case模型方面，我们于9月11日第一次组内评审use-case模型，在初版use-case模型上对需求的功能增加了私信的功能，以方便用户实时联系；9月23日组内评审决定增加数据分析的功能模块；9月30日再次组内评审使用PowerDesigner绘制的use-case模型，补充加入教师对课程的管理功能。  2. 我们于9月23日进行了《vision》文档初稿的组内评审，修改了《vision》文档中一些表述不清晰、不确切的地方。并在9月28日老师上课点评过后再次对《vision》文档进行了修改。  3. 我们于10月9日进行了《软件需求规约》和界面原型的组内评审，修改了《软件需求规约》中表述不清晰、不确切的地方，并检查了界面原型是否达到了预期的效果。 | | | |
| 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. 在确定产品名称方面，产品名经过多次修改，对文档进行了反复修改，对文档中的产品名进行替换，不需要返工。  2. 在项目需求以及use-case模型方面，在use-case组内评审的过程之中，遇到了功能阐述不明，主要功能缺失等问题，这些是我们组内对已有功能需求以及可能的功能需求进行分析探讨得出的。在一致表决同意过后，会对已有的use-case模型的功能做出补充或者修改。每次变更都可以在讨论之中完成，因而不需要返工。  3. 在完成《vision》文档的时候，对产品可靠性的设定不到位，对系统可靠性要求过低。  之后我们对前景文档中的可靠性的描述进行了修改，并准备学习新技术保证更高的可靠性。  不需要返工。 | | | |
| 经验和教训：  1. 在《vision》文档初稿中我们对于系统的可靠性设定过低：对于一个需要频繁使用的系统来说，可靠性为一个月内95%的正常使用时间偏低，而本产品作为一个全国学校师生都会频繁使用的系统，其可靠性应做更高要求，至少需要达到99％的正常使用时间。  2. 在设计界面原型的时候，由于前期对于功能性需求的理解和界定不清楚需要返工，导致一部分界面原型的设计迟迟不能开始，险些逾期。这给我们的教训是对于两个互相有依赖的任务，应该尽早开始被依赖的那个任务，并且争取提前完成，留有一个对于成果进行更改的时间余量，不然会对之后的任务进度产生负面影响。  3. 对于有些需求和功能组员之间没有达成很好的共识，导致了一些任务需要返工，降低了效率。因此对于一些有必要的文档不能偷懒少写或者是不写，因为文档是一种很好的规约方式，可以记录下组员之间达成的共识，方便在进行之后的任务过程中参考，提高了团队开发的效率。 | | | |