

软件工程个人总结
郑嘉诺 2013010383 环 31

一、前言

其实一直因为对于软件开发过程中的团队合作不熟悉所以很想上这门课，而大二大三两年都与别的课程冲突而未能实现。本学期发现了其可行性以后果断扯上了一些的人组了个队伍，虽然途中遇上了很多问题.....嗯，主要是 OpenGL 部分完全学习不能使我们抛弃了 Qt 开发环境转而选择了 Unity。导致了我们必须重新开始。所以能够顺利完成，我真的很感谢各位队友的鼎力相助。否则.....我可能会比较忧郁。

二、个人开发感想

以前我就一直憧憬能够进行一次团队开发合作。这次终于实现以后，我是十分满足的。也许确实我是喜欢进行程序开发吧.....在保研期没有在环境学院继续而是选择去了软件学院，我现在更加自信是一个正确的决定了。

在开发过程中不断地出现过各种小问题，不过和我想的一样，基本上都是稍微花点时间就能解决的.....除了 OpenGL，我之前完全不了解时，以为是和各种可视化编程一样的简单易懂的开发方式，直到我有了更多了解后才发现，根本不是能够在短时间内克服的。相对应的偏向于在高层编程的简单好用的 Unity 让我现在确实喜欢上了它。不愧是现在最火的 3D 游戏引擎，可以说实至名归。

我很难说自己是一个合格的组长。在开发的前期时我经常激励组员，每隔 3 天左右就打听一下进度，说明一下下一步的工作目标，然而进度十分缓慢，基本上 3 周才完成了预计 1 周能够完成的工作.....我必须重新评估我的进度计划是否合理。可能就是因为在这个方面受挫，在后来我变得没有那么催人了，有的时候有点“做了自己该做的事情就好了有点不想管整体了”的感觉...，除了每周固定租用团队研讨间给一个下午的集中编程时间外，没有做太多别的了...然而各位组员的效率反而都上来了。也许果然不应该催太多吧.....我还在继续反思。不过说到底反正我又不给工资，激励效应当然没有时间大啊！（捂脸）

三、项目开发经验教训总结

第一：程序结构专门控制很重要。Unity 程序的开发过程中，因为开发匆忙，没有余裕像 Qt 程序那样慢悠悠地设计好程序结构再慢慢完善细节，而是简单地直接就往上面写功能然后硬性拼接，导致的结果是程序可读性差，很难说出程序的运行顺序等，出了 bug 也总是四处撒网写调试语句才能找到，更新新功能容易导致旧的程序出错。因为 Unity 无法写单元测试，因此出错概率更高。如果控制好了程序结构，再严格以此写程序，也许就不会因为 Unity 协程结算顺序而必须将逻辑结算显式写在每个动画模块的下面这种不易扩展的方式了吧。

第二：一些开发过程中的基本原则也很重要。例如，不要修改他人写的代码，而是通知他进行相关修改。有组员就有过擅自修改了其他人程序代码而没通知，分支合并时出的 bug 数目的数字，我觉得我一生难忘。又如，Git 的合并冲突问题，如果不管理它而直接强行 commit，导致过好几次 UnityScene 无法识别，版本莫名其妙回退等诡异事件。如果每次都都将冲突解决了，没有残留的话，是没有出现过这样地事情的。

第三：团队的工作效率是分重要的。有很多周我都想过“如果之前能更快一点的话”等这样地想法。不过说到底，想法很简单，激励他人工作效率却并没那么简单就是了...这是我需要学习的。