

软件过程改进心得体会

13331202 莫冠钊

一、个人贡献：

- ①参与前期项目数据库搭建的讨论。
- ②参与页面功能逻辑、交互设计方面的讨论。
- ③参与项目中“活动详情”页的大部分样式架构的搭建，以及部分逻辑的实现比如筛选活动参与人员、取消筛选、控制图片的显示。
- ④在项目改进的过程修改了部分bug，如筛选成功后每个人的_id都变成了第一个人的_id，改进前端样式，使交互更加友好。

二、课程心得体会：

还记得第一次完成的作业叫做环形菜单，当时只有最终成果demo的展示，没有《web2.0程序设计》那样一步一步地指导提示。花了老师给的时间（8小时）的一半，做完之后给我的感觉就是我在四个小时在探索web基本样式的精华，这一过程我无比享受。在后面的学习，用JavaScript操作其他页面的DOM对象、面向对象JavaScript及闭包原型、异步编程等等知识，这些东西让我深深体会到web的精妙之处，使我对web应用产生了浓厚的兴趣，我觉得能够通过自己的代码实现这样的效果是一大成就。

然而，我这样的想法是不对的，真正有价值的工程师应该是把大量的时间放到自己的Thinking上，减少Code时间，最大化地从自己的产品中展现出自己严谨的编程思想。这也许算是王青老师上课所讲的“面向思想编程”的部分内容。但是原生的web前端语言在语法上又有一定的限制，不能随心所欲地用简单的语句来实现对应高效而又严谨的代码效果，所以这时候就必须依赖于友好的编程性工具。学习工具的周期比较短，而且学习工具就是在接受一种新的编程思想，因为每种工具的产生都有它的存在价值，都有它所解决的主要问题。本课程中所用到的Grunt能够更好地帮助我们管理整个项目代码，自动刷新页面，及时反馈错误，并且它还能编译LiveScript、CoffeeScript、jade、sass等等，能够十分友好地展现出项目的TOP思想。

紧接着，王青老师所教给我们最后的一个知识是SPA（single page application），这个是Web应用的一个先锋代表。SPA能够将浏览器当作是一个客户端，前端与后台职责分离、架构清晰，前端与后台单独开发、单独测试，避免不必要的跳转与重复渲染是SPA提高用户体验的一大重要因素，它将是Web应用的一大发展趋势。它所依赖的技术有Ajax、JSON等等。不过Meteor框架能够帮我们免除这些技术问题，我们只需要对数据库进行相关的数据操作，Meteor将自动完成中间的数据交互，同样大大减少了code时间。

“软件过程改进”这六个字更多是体现在课程设计项目开发中，无论是哪一个部分，项目策划，交互设计、视觉设计，架构搭建，项目开发，交互逻辑，后期维护等等，软件在开发的整个过程都会贯穿“改进”的思想。团队合作固然重要，但是团队在合作的过程更重要。leader怎么引导整个团队正确地思考，围绕问题的核心点出发，从用户体验上考虑，从技术难度上考虑，如何取舍，如何高效，如何降低出错.....这些小小的细节一步一步地促成改进的这个过程，而在这个过程中同样也离不开TOP思想。最后我们的项目SEEK也顺利地落下了帷幕。

别的课程教给我们的都是一些知识性东西，而《软件过程改进》带给我们的则是一个思想，知识是学习的一个产物，而思想则能够影响学习的方法，从而获得更多更深入的知识，十分感激《软件过程改进》。

三、课程中学习到的技术能力：

- ①CSS预编译器：Sass/Css
- ②项目构建工具：Grunt、Meteor
- ③后台开发：NodeJS
- ④其他的Web相关技术：ExpressJS、LiveScript、CoffeeScript、MongoDB。