

The logo for 'Activitee' is displayed in white text on a solid orange rectangular background. The word 'Activitee' is written in a clean, sans-serif font, with the 'A' being slightly larger and more stylized than the other letters.

期末项目汇报

Activitee——活动信息展示平台

下午班第三组
2015年6月16日

项目概要

这是一个校园活动信息收集平台。用户可以通过此平台浏览和发布校园活动信息。

功能介绍

请看视频介绍：http://v.youku.com/v_show/id_XMTI2NzkyMDk4MA==.html

项目部署服务器地址：<http://120.24.211.137:5000/>

改进记录

第一周

进行Course Project的初步设计，撰写Course Projects Proposal。

第二周

1. 充分分析、挖掘活动发布方的需求，例如发布方无法做到针对性推送活动信息，我们通过“用户订阅机制”来满足其需求。
2. 完善 Course Projects Proposal。

第三周

1. 进行原型图的绘制
2. 建立数据模式、模型
3. 整理各个页面的渲染参数

第四周

1. 进行设计图的绘制
2. 后端进行页面数据渲染、数据库交互

第五周

1. 重构登陆、注册模块
2. 后端各个controller的编写（页面渲染、数据处理、功能实现）
3. 初步实现图片上传
4. 初步完成主页功能模块：活动筛选、排序、分页，完善前端交互，动态加载活动内容。

第六周

1. 各个页面前端的编写
2. 完成主页的轮播（slider）的实现、活动标签的展开和隐藏、回到顶部按钮、分页功能。

第七周

1. 实现头像、活动封面上传后裁剪功能
 2. 实现嵌入地图功能
 3. 实现各个页面的权限控制管理
 4. 测试各个功能模块，修补发现的bug
 5. 设计活动和用户数据，充实网页内容
-

组员贡献和课程心得

刘焯鹏 13331167

担任组长，负责设想整个项目创意，设计需求、数据模型，负责首页、网页头部和脚部前端编写，以及辅助其他成员完成各个页面的前端部分，统筹规划全局。

这学期软件过程改进课让我收获良多。作为MWL的一员，软改课上前期的Lab我是做过的，但重新做一遍有不同的收获，温故而知新。而中期开始接触express框架，让我彻底喜欢上了nodeJs，而学习grunt等自动化工具也提高了我的编码效率。虽然每周的Lab难度都很高，花费的时间甚至比其他科目加起来的时间还要多，但同时收获也是非常大的，为后期的课程项目奠定了坚实基础。

而让我收获最大的是，课程后期小组共同完成开发的课程项目。作为组长，我不能仅仅只会编程，还需要统筹全局。这个阶段我们不能再停留在完成作业的层次，而是完成一个产品。从创意、需求，到每个细节的设计，再到分工，以及每个人工作的落实。这些工作对我来说是很大的考验。途中也遇到很多问题：比如组员之间沟通效率不高，组员不熟悉github导致项目代码管理混乱。在这过程中，我也逐渐熟悉了一个项目逐步向前推进的过程，虽然还很不专业，但这也是软件过程改进的教学意义所在吧。

邝嘉豪 12330152

负责登陆、注册后台模块编写，以及难度比较高的邮箱验证、图片上传裁剪功能，负责相应页面前端编写。

- 当初选这门课是因为正好选定Web开发作为自己的兴趣方向(另一个原因是实习季将至，需要定一个目标岗位，选得也是Web前端)，之前选过王青老师的公益软件与协同设计，印象比较好(虽然每周惯例被批不过确实有收获)，毫不犹豫地选了这门课，因为学分限制，最后是用过选课申请表选上的，不过一学期下来确实没让我失望，作为工程类(而非研究类)课程，算是质量比较高的课，包含以下几点
 - 没有Web基础的讲解而是直接作为一个门槛，这很好。个人理解：Web技术 = 技术规范 + 解决方案，前者只是属于规范层面的内容，并不能令人在思想上取得很大收获，而且更新非常快，不适合作为课程内容介绍，更适合自学；后者，设计解决方案，我觉得是个极具挑战性的工作。WHY? 做Web开发有个特点，自由度极高，在开发过程中会使用大量的hack解决问题，而各种做法都没有真正的best practice，更多的是经验之谈以及根据实际情况的即兴创作。什么是hack? 这里给个中文翻译：奇技淫巧。我很难对此下一个完整的定义，不过例子有很多：
 - 如何在GET 一个页面的同时存储服务端提供的一些数据？
 - 1 假设是一段文字，直接输出到一个div然后hide即可，可以通过DOM操作取得数据
 - 2 假设是一段包含各种符号的字符，可能破坏文档结构，那么对敏感字符做转义即可(当然读出来的时候也要做反转义)，更简洁的做法是输出到一个注释结点(一般的文本数据不会包括注释的起始标志，当然如果要做得严谨的话还是要转义)
 - 3 假设是二进制数据，那么可以建立字符与字节的双映射，这里可以自己设计映射表，也可以用一些现有的规范
 - 如何实现页面部分更新
 - 1 前端fetch数据，然后做字符串拼接得到html片段，最后做DOM操作完成更新，简单粗暴但可读性糟糕
 - 2 前端fetch数据，然后使用前端模板引擎渲染得到html片段(本质上与1相同)，比较优雅但是要引入前端模板引擎，另外一个问题是模板数据来自哪里，答案参考上一个问题
 - 3 前端fetch html片段而不是pure data，渲染工作交给后端，优雅且简单，但是会增加网络IO，而且接口扩展性复用性下降(只能用于Web前端而不能用于其他客户端比如爪机app)

-
- 扯得有点多,就此打住
 - 老师对项目开展流程的讲解和监控都非常到位,指出各组开发进程的问题时尤为犀利,直戳要害,没有放过任何潜在的问题(至少包括我能看出来的),不得不服
 - 介绍了一些业界流行的工具,虽然说思想远远重要于工具,但是了解前沿工具对拓展视野也是重要的一环
 - 以上是对课程的评价,下面说说大proj的个人心得吧.首先不得不大赞一下我们组的设计师,作品虽然算不上惊艳什么的,不过作为软院的学生真的是极大超出我的预期;另外组长也是比较负责,算是靠谱.总的来说组员的积极性很不错,让我有了一次挺愉快的团队开发体验.说完组员再说说我自己吧,因为这学期课比较多(高达15分的选修),导致不能如愿地在这门课投入大量精力,处于半划水状态,有点可惜.不过作为自认为思想和技术都不错的hacker/coder,还是有不少突出的贡献,比如组件化的文件结构就是我的建议,大大改善了开发中的舒适度,还有为项目定制的用户验证组件,邮件验证功能,图片上传/裁剪组件,比起直接使用一些现成的组件,学习成本极低,而且因为是DIY,可以完全为项目量身定制.
 - 先扯到这里.

叶嘉祺 13331314

负责关注、参与活动、评论活动等功能模块编写,负责相应页面前端编写。

其实当初选这个课的时候,并不知道这是干嘛的。因为学院的选修课本来就少,所以方式选的人是很多。之后听说出了一份通知,需要有web基础我就对这门课感到更加奇特。

然而,这门课真的很难。前大半学期熬夜写软改到三点是经常的事情了,因为那时候只上了web2.0,只有一点点基础,所以上起来非常吃力。而且这门课,说实话用到最多的语言就是javascript,这对于我这种钟爱c/c++的人不太适合,特别是那种动态继承机制,让人很吃力。然而吃力总是有好处的,就这样,软改这门课学到了很多很多东西。让我印象最深刻的就是top思想,这种从上而下分解问题的思考我也曾有过,但是总是想到一半就会卡住,做不下去,然而,现在我懂得了用工具,图像来帮助我,这个在其他课程上,比如数据库的课程设计上也有很好的应用(这种思想很珍贵)。还有就是深入思考的能力,最让我印象深刻的是老师举得那个审核的成本的例子,其实有些东西从表面想去都会有一种给人的错觉,但是,如果深入的思考,可能就有不同的结果,或许思考的结果会很惊人。

到了后面阶段的课程设计,我从来没有做过这么大的项目。哦对了!在github上面的开发学习,协同工作这种工具的学习也是让我进步很多,思考很多。9个人一个小组同时开展工作,从定题到实现,解决各种问题,都是一个很有挑战性的过程,我们在中途遇到了非常非常多的问题,然而我们总是能在风雨之后找到那么一抹阳光,把问题解决掉,最终我们呈现出来的效果,我自己都惊呆了。这个项目有很大的价值,也许它只是一次真实的作业,但我觉得这是一次很珍贵的经验。

到了这个时候,我终于有点理解了软件过程改进的含义了!之前的作业基础中就已经有少许这个倾向了。到了后来我们的课程设计就更加是了,我们的项目经过一轮一轮的迭代开发,不断改进,得到了完善,我们在这个过程之中的思考,才是改进的核心!

这门课给了我很多的经验!最后谢谢辛苦的王老师,和Ta们!

林在华 13331163

负责活动筛选、排序、分页、地图等功能模块编写。

我觉得软件过程改进是一门能够让人学习知识又能锻炼学习能力的课程。课程具有一定的深度,要求有web2.0基础,否则将会受到很大的阻碍。课程按计划快速推进的做法能够让我觉得每周沐浴在新的知识

里面，每周能学到新的知识。在这种快速紧张学习的氛围下，个人的快速学习能够得到很大的锻炼。当每周在紧张的时间里通过看大量的文档、学习大量的新的知识完成课程作业后，个人会变得自信，也会获得满足，这在一定的程度上激励着个人下次继续认真完成作业。在这种良心循环中，吸收海量的知识，锻炼学习能力，提高学习兴趣，自然会越学越快，越学越清晰。

但是，由于课程节奏快，学过的东西很容易被遗忘，所以我觉得思想是最重要的，学会了思想和分析问题的能力，自然就容易上手，通过学习大量新的知识来锻炼学习能力，无疑是最快的捷径。每周三到周六都会浸泡在各种文档里面，形成了一种习惯，养成了学习新知识阅读文档的好习惯。

课程给予我们组队完成project的机会，让我们能够运用课堂上学到的知识和技巧，应用于实际开发中，模仿实际开发，按着流程一步一步走下去，在过程中学习，在学习中改进。还记得每次课后大家都会聚在一起讨论很长时间，讨论创意中的问题、讨论开发中的问题等等，形成头脑风暴，刷新脑资源，更新对创意的认识，学习别人擅长的知识。还有每周二晚上的会议，我们能够展示自己一周的工作，极大的促进队员的工作进度，使得队员对项目有更高的热情。

我们的组长非常认真负责，只要我们有问题都能询问他，他也很乐意和很负责的解决，组长非常优秀，他能够让我们很欢乐的对待项目并且调度我们的积极性。我们组自产的美工也非常棒，不辞劳苦的承包了Activitee的所有UI设计。还有大三的师兄非常有实力，能够解决一些非常难的问题。

这个课程让我学到了知识，见识了开发，学习做项目，认识能力强的同学，打下牢固的基础，留下美好的回忆。

郑仪旺 12330423

负责个人设置页、用户信息编辑、更改密码等功能模块，负责相应页面前端编写。

在最初的时候，我还以为这门课就是学前端，调样式，每周实现一个作业完成一项功能。不过随着课程的进行，从软件设计那节课开始，我开始认识到之前的学习的都是基础，更重要的是后面所学的应用设计，正如老师所说的如果不能做正确的事情的话，事情做得再好也没有意义，而且要按照一套流程才能避免走错路弯路。

最后我们小组做的是活动发布展示平台的网站，虽然最后进度稍微领先于其它小组，不过很大的原因是占了选题的优势。即便如此，我们在开发的时候依然遇到了不少问题，比如在数据库的设计方面，我们最开始的时候没有完全确定活动和用户包含的字段，给开发带来了不便，在后面每周的进度汇报展示通过老师的点评明白其实这个问题是因为我们在进行原型设计的时候把任务直接交给了小组的一个成员，后来虽然出了好看的界面，但是因为没有团队一起坐下来好好进行需求复审，用上一些角色扮演或者用例卡片的方法分析，及早发现必要的改动。

这门课程里的小组成员都很认真积极，组长也是非常负责，使用github的第一次团队协作开发体验也非常地好，组长还总结出最大可能避免merge时的冲突的pull方法，就是每次pull给组长前先把自己本地的更新到最新，然后才写入本次改动，这个方法在后来我也用到了自己另外一门课程的项目中，能够减少组长在merge解决冲突时花费的时间和精力。

最后不得不说的是我对TOP的感觉，感觉依然停留在最初的理解面上，即写出更加易于阅读理解的代码，但是我总觉得这个更多的是受限于所选择的语言，比如我们选了jade, ls, python, 写出来的代码肯定比html, js, c++赏心悦目得多。尤其是到了项目后期，似乎完全没有考虑到这个理念，只是一心地完成某项功能，虽说也有注意代码风格，但是与以往的代码并没有太大的区别，更别说写出那种虽长但是又容易理解而且一行就能完成比较复杂功能的代码了。有时为了达到前面所说的目的就实现这么一两个辅助函数，但是往往重用的机会不大，总感觉这样没有必要；有时又为了提高函数的复用的机会设置了一些参数使得函数不够单纯，其实自己更加欠缺的是设计恰当的辅助函数的能力，这要在以后的实践中多多锻炼。

再总结一下我在这门课程上的收获: 我不仅接触了web开发的前后端, 第一次体验了使用github工具进行团队开发, 同时还简单地学习了软件开发的流程规范, 并认识到其对保证我们项目的完成的重要意义。

林泽明 13331164

负责“我发起、参与、关注的活动”页面功能模块编写和前端编写。

本学期我选了王青老师的软件过程改进课程。经过一个学期的学习, 我学到很多知识, 对于软件过程有了新的看法。我觉得软件过程改进是web2.0的进阶版, 因为这门课程需要有学习web的基础, 而且这门课程的难度不低。

软件过程改进这门课分成2个阶段, 前一个阶段是单独做几个作业, 用到了不少新知识, 我从中也学会了用express js和meteor的框架。我认为熟练掌握web的基础知识是非常重要的, 因为万变不离其宗。比如回调函数, 构造子等, 是很重要的知识点。而当我们的基础够扎实时, 做project就会得心应手。

后一个阶段是团队合作做一个project。我觉得这个阶段有几个点是非常重要的: 一是要有创意, 二是要符合市场需求, 三是要发挥团队合作的精神。我们组是做一个关于校园活动查询和报名的project。一开始遇到不少问题, 比如校园活动的来源。后来经过老师的建议和组员的讨论, 才进入正轨。我们组的进度比较快, 主要是因为组长认真负责, 组员之间保证交流。最后我们组做出一个基本成型的项目。虽然还有不少瑕疵, 但是却是我们整组努力的结果。后续工作我们和完美这个项目。

我认为在本学期的课程中, 软件过程改进对我的影响最大, 收获最多。课程结束后, 我会继续保持一颗为软件开发而奋斗的心, 致力于成为一名优秀的软件工程师。

张国伟 13331334

负责发起活动、编辑活动功能模块编写和前端编写。

在这门课程上, 我学习到了各种Web领域的技术, 像CSS3的酷炫效果、Javascript的坑和hack、ExpressJS、meteor等框架的使用方法和发展历程, 虽然这些加起来都只是一点皮毛; 同时也对异步操作有了一定的理解, 如何在异步的情况下控制执行顺序、以及怎么解决由此而来的callback hell; 还有就是通过嘲笑反人类的设计, 了解到什么才是友好的易用的设计。当然还有这门课最强调的, 软件开发的过程, 自上而下的程序设计方法, 项目的设计, 原型的重要性, 不敢说掌握了老师所传授的全部, 也算是有所长进吧。每周布置的任务都很有挑战性, 迫于课业所困没法花太多时间去研究, 以至于有时不能在规定时间内完成, 即使完成了也不能很好地掌握, 罪过罪过。另外老师的Live Programming真是让人眼前一亮, 不得不佩服。总的来说, 一分耕耘一分收获, 痛并进步着。

刘健坤 13331170

负责管理员审核活动功能模块, 负责相应页面前端编写。

在课程的最后一次实验中, 我负责的是admin该页面的相关代码。在项目刚开始的时候, 我们确实很迷茫, 这时候组长搭好了整个框架, 使得每个人的分工明确, 在我们有美工的情况下, 项目的进展便进展得很快。

这次实验中我深深的体会到有队友, 有大腿的好处, 很多情况下的代码debug和整合, 都是组长完成的, 而美工又有专人负责, 也不需要考虑太多, 只要安安心心做好自己的本职工作即可, 所以这次项目我个人是相对来说比较轻松的, 而且我个人负责的页面也是比较简单的, 对于整个项目的贡献率也相对低了些。

而对于自己完成的那部分，难度主要是在和其他组员保证尽可能一样的格式，包括代码格式，主要是函数的命名等，虽然这个会比较麻烦，但是作为一个大项目，我觉得统一格式是很必要的。

对于整个课程，感觉自己的收获是很大的，比如了解了很多新的技术框架，还有一些模式，mvvm等，这些关于web2.0的东西都是很诱人的果实，比如对于这次的课程实验，我们就用了node.js和mongodb的数据库，他们用起来很轻便，使用起来很是简单，而且不需要花很多时间就可以熟练的掌握，这应该就是web2.0的魅力吧。

这次项目带给我个人的乐趣还是蛮多的，比如Mac系统妖孽的美化功能，我在windows上调出来的效果始终不够好看，然而跑到Mac系统上就迷一般的变得优雅很多。可能很多是字体方面的问题吧，在运用了less之后虽然好了一些，不过Mac的确实比Win的好看不少。

其他小组的项目也是很有创意的，但我更觉得我们小组的创意更赞一些，不过这个更适合用Java做成个App而不是一个网站。当然，因为我们课程的主题是web，做网站就更加符合主题了。

总之，自身的收获和得到的东西还是蛮多的，包括经验技术方面，而且也认知到了自己的不足。毕竟个人还是没有能力去搭一个如此的平台或者网站基础，同时也认识到了自己需要努力的地方，还是收获满满的。

谭笑 13331236

担任设计师，负责所有设计图的绘制。

起初是因为自己对 web 开发的兴趣而选了这门课程,然而老师在第一节课就 说这门课程很难,劝我们量力而行,可以退课。的确我们从第一节课就体会到了 这一点。第一节课的作业,是听上去很简单的 css,但是为了实现尽可能完全一致的样式,仍然付出了无数的心血。而此后的作业也越来越难,尤其是到了要求 我们用 TOP 的思想来完成作业的时候,不仅要求我们有着很高的自学能力与自觉能力,更要求我们深刻地理解 TOP,真正地意识到 TOP 给我们带来到的好处。

到了课程的第二阶段,小组 project 部分的时候,我则更加深刻地体会到了“软件过程改进”这六个字的含义。一个软件生产的过程中总是会产生各种各样的问题,比如课件中的那张图。我们小组设计自己的 web 应用的时候也遇到了 不少的问题。首先是用户需求与我们理解的出现了一定的偏差,在老师提出观点 之后,我们便针对这些观点逐一进行分析、讨论,一点一点地完善我们的 proposal,终于在第二周,正式开始设计我们的 Activitee。这次的 project 中,原型图和设 计稿是由我负责的。在上了课之后,又查阅了一些相关资料,了解到一个软件产 品的开发中,原型是很关键的一步。于是我就根据我们小组分析出来的用户需求,先是绘制了原型图。其实我没有想到的是,下午其他两个组起初都忽视了这一点,而老师对我们的工作进行了肯定,这让我们组都感到十分鼓舞。到了 UI 设计部 分,我也不是专业的设计师,但是为了更加了解相关知识也临时看了几本与 UI 设计相关的书就投入工作了,实际还是遇到了一些问题并且被老师吐槽了。除了 原型图、UI,就是前后端的开发了。更改了代码如果不及时 pull request 然后 merge 的话,就会引发许多冲突或者其他问题,这也体现了小组合作其实并非想 象中那么容易。但总算我们组尽力地完成了这个 project,实现了绝大部分功能。

总的来说,这门课程真的很难,但是真的让我学会了许多其他课程学不到的 东西,而我会在今后的学习与工作中,努力将所学知识贯彻。
