

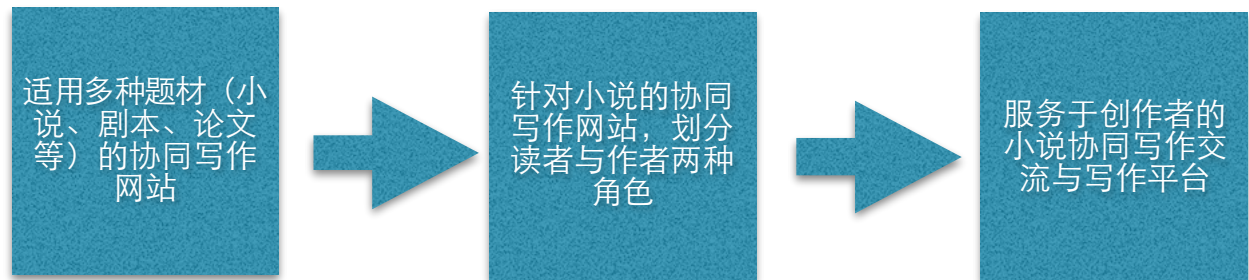
Bookhub项目报告

一. 简介

Bookhub，专注于协同写作的小说网，为小说创作者提供一个方便易用的交流与写作的平台。

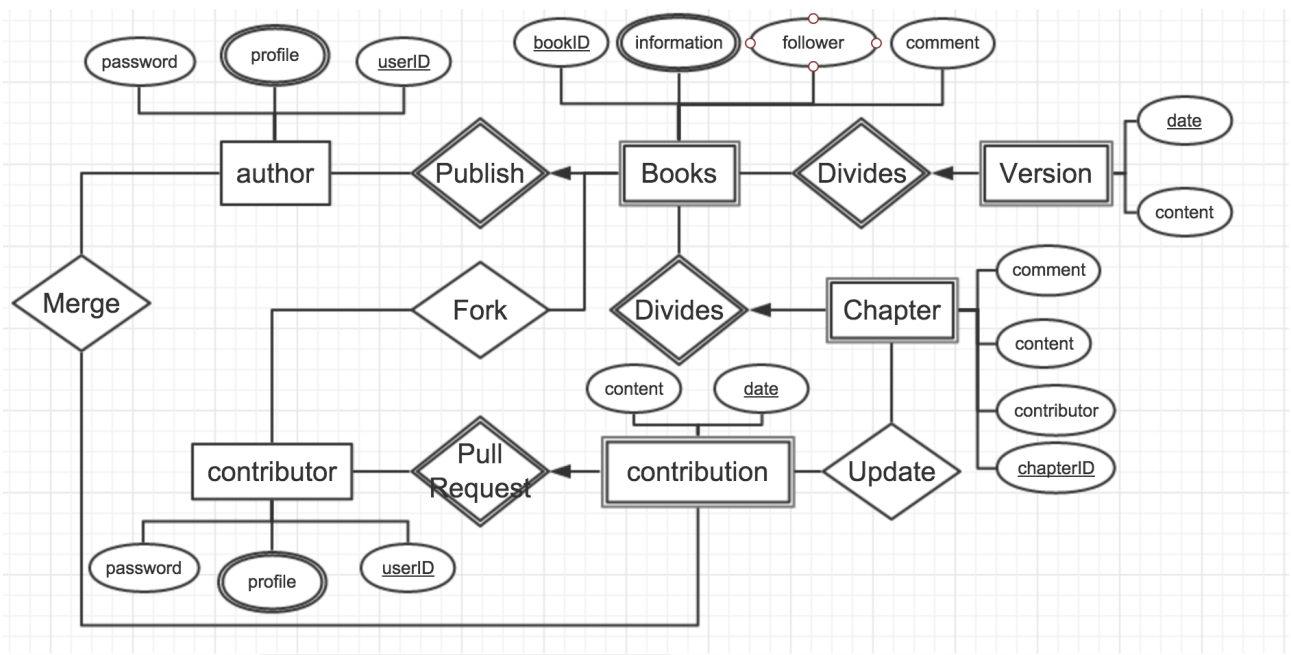
作者编写的小说一旦完成，可自由投稿到其他小说网站(遵循一定的利益分配规则)，以鼓励各创作者参与其中，贡献更好的创作想法，达到共赢。

二. 立意改进过程



三. 数据模型

• UR模型图



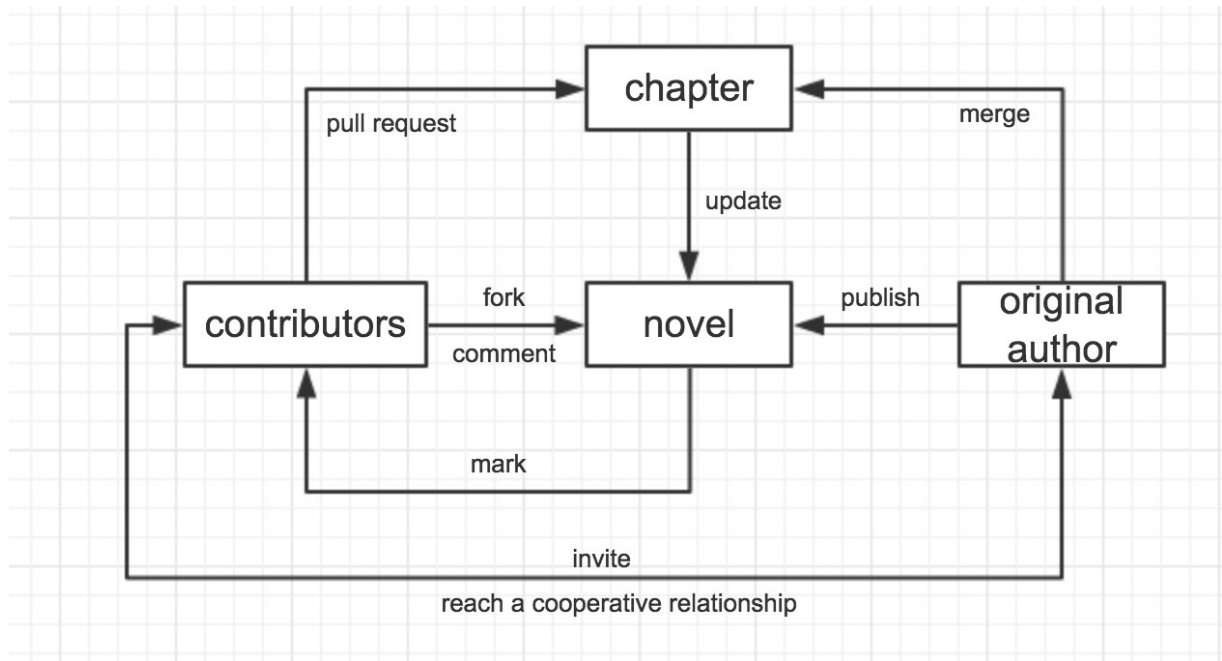
- 改进后的User数据模型实例

```
Meteor.users =  
  username: String      #用户名  
  password: String     #密码  
  
  profile:  
    pseudonym: String   #笔名  
    introduction: String #个性签名  
    gender: String      #性别  
  
    fork: [{  
      NovelId: String    #参与协同写作的小说  
      createAt: Date     #小说ID  
      #fork日期  
  
      chapter: [{  
        index: Int       #记录参与该小说的哪章节  
        summary: String  #章节  
        content: String  #修改声明或故事梗概  
        #修改内容  
      }]  
    }]  
}
```

- 改进后的Novel数据模型实例

```
Novel =  
  authorId: String      #作者标识  
  authorPseudonym: String #作者笔名  
  name: String           #小说名  
  type: String           #种类  
  summary: String        #小说简介  
  pv: Int                #阅读量  
  follower: Array        #参与该书协同写作(fork)的其他创作者  
  createAt: Date         #创建时间  
  curVersion: Date       #当前版本  
  lastChapter: Int       #最新章节  
  
  versions: [{  
    createAt: Date       #版本，以时间戳为标志  
    chapters: Array      #创建时间  
    #包含的章节，具体与以下相同  
  }]  
  
  novelComments: [{  
    reviewerId: String   #小说总体评论  
    content: String      #评论者  
    createAt: Date  
  }]  
  
  chapters: [{  
    index: Int           #小说章节为基本单位  
    title: String        #第几章  
    summary: String      #标题  
    content: String      #梗概  
    contributorIds: Array #内容  
    createAt: Date       #修改者  
    modifyAt: Date       #创建时间  
    #最近被修改时间  
  
    comment: [{  
      reviewerId: String #章节评论  
      content: String    #评论者  
      createAt: Date  
    }]  
  }]  
  
  pullRequest: [{  
    contributorId: String #贡献者提交的修改请求  
    summary: String       #贡献者  
    content: String       #修改声明  
    createAt: Date        #修改内容  
    isMerge: Bool         #修改提交时间  
    #是否被原作者采纳  
  }]  
}
```

四. 功能模型



- **fork:**
fork小说，表明参与到该小说的协同创作过程，也才有修改内容、提交合并请求的权限。
另外，每本小说只有最后一个章节开放给其他创作者协同参与（避免了前后章节的修改冲突）。
fork之后，在自己的创作页面中的“编辑的小说”标签页面显示该小说。
- **pull request:**
当贡献者完成了自己的修改内容，可以提交合并请求给原作者。
- **merge:**
当有贡献者提交请求，首先显示修改声明，如果原作者感兴趣，可以点进去阅读全文并可采纳。
一旦原作者采纳了贡献者的提交请求，更新至自己的小说章节，并且显示该贡献者。

五. 技术选型

- Meteor 单页面与多页面结合。
- 使用Jade、css、Javascript实现。
- 前端框架使用了Meteor的 semantic-ui packages。
- 图片上传（头像、小说封面）使用了Meteor的 CFS packages。

六. 小组分工

小组分工贡献

角色	姓名	学号	贡献度	负责的项目部分
组长	肖燕浩	13331290	17%	1. 每周的项目进度展示 2. 组织项目以及分工、维护Github 3. book模块和contributorEdit模块 4. fork模块 5. 项目后期的完善和报告
	胡江川	13331078	15%	1. 前端框架的选择和搭建 2. user模块和authorEdit模块 3. pull request 和merge 模块
组员	韩龙粤	13331067	13%	1. 网站logo 2. 图片上传（书封面和用户头像）
	郑力枪	13354467	13%	1. bookIntroduction模块 2. 数据模型
	杨振杰	13331311	13%	1. home模块 2. 准备项目进度展示PPT
	牛仁鹏	13331203	12%	read模块
	苏永健	13331227	7%	数据模型图
	王文茏	13331256	5%	参与讨论
	陈纪云	12330018	5%	参与讨论

七. 小组总结

小组个人心得总结

姓名	学号	心得总结
肖燕浩	13331290	<p>一个学期下来的课程学习与实践，到这里要画上句号了，但是课程之外的学习与实践似乎永远都没有尽头。软件过程改进，核心在于改进，也只能在接近尾声再回首时，才能深有感触。除了对于技术层面(语言、框架)和项目管理工具（github、grunt）的快速学习，带给我更多的，是TOP编程思想和项目的改进过程。</p> <p>TOP编程思想利用了Livescript语法简洁自然的优势，是强调以人的思维为主导的编程方式。将想法明确地用近乎自然的语言展示出来，在语言之上处理逻辑。这样一来，代码不再是死板的语法，而是活灵活现的思维的体现。编程过程不但保持了模块化，而且还紧紧跟随于人的思维活动。</p> <p>后期的小组项目过程中，可谓是举步维艰。在进展中我们想得太快，做得却太慢，大多是我们忽略了过程改进中所必需的步骤。再者我们的项目创意有些特别，也注定了要比其他小组走更多的弯路，然而在初始阶段我们组内甚至还没有达成共识。一路走来不断倒退回去重新再来的教训，也让我们懂得了软件过程的改进是不能一意孤行、空谈冒进的，更不能在没有越过技术风险口之前就开始项目的开发。</p> <p>另外，身为组长的我，从中也认识到搭建项目、组织架构的重要性。开发项目阶段时，组织团队协作和分工成了我每一天的苦差事，因为难以避免冲突与不协调。然而，这也成为我人生中第一次组织团队开发项目的宝贵经验。</p>
胡江川	13331078	<p>Thinking Oriented Programming，我觉得这句一直挂在老师嘴边的训言总括了软件过程改进课程方方面面。编程作为人类的一种思考过程，需要变得简单并且少犯错误，我们需要放更多的精力在程序设计本身上，而不是其他零碎的事物。从一开始写的CSS3环形按钮，到可排序表单，到ExpressJS，到Meteor。由简至繁，我看见了web技术的蒸蒸日上，以及它为程序员提供的越来越大的帮助，编程变得不再是煎熬而是一种享受。唯一遗憾的是，我没有办法放慢脚步去细细品味这些技术。回过头来才发现，自己很多时候是为了完成作业而做作业，这门课程的精髓却在不经意间忽略。课程进度实在太快，需要学习的信息量远远超出我的预想。的确，这是一门十分具有挑战性的课程，但我喜欢！我喜欢那种学习到新知识的成就感，喜欢自己的手指不间断敲打键盘的快感，更喜欢课堂上新旧思维碰撞产生的火花。学习不就是先苦后甜的过程吗？</p>

姓名	学号	心得总结
韩龙粤	13331067	<p>* 一个学期的软件过程设计的课程渐渐走到了尾声，在回顾整个学期的过程才发现自己其实也是做了挺多东西的。</p> <p>* 学期的前半段是疯狂学各种新的语言、新的框架、新的思想：纯CSS实现环形菜单、利用JS做表单排序、AJAX、TOP编程思想、使用expressJS、meteorJS实现简单作业系统，中间还有包括Jade、livescript、Grunt的快速学习与使用。</p> <p>* 在学期的后半段，小组集体完成了一个协同写作的项目，深刻的体会到了“改进”二字的含义，并且经历了小组合作的种种体验。</p> <p>* 虽然一开始对短时间内学如此多新东西感觉不是很理解，但是大量学习之后也发现学新东西并不是最重要的，学会如何在短时间内学习新东西以及学到那些编程的思想才是重要的。</p> <p>* 另外获得的经验还有要善于使用工具，在不同的需求下合理地选择不同的技术实现方式才能做到又省事又省力。</p> <p>* 在最后不断立项又被打回修改的过程中，体会到一个真正项目的立意、目的、功能等等启动工作如果不做好，后续的工作会举步维艰。</p> <p>* 最后谢谢老师，谢谢TA，谢谢组员，这个学期因这门课而变得更充实！</p>
郑力枪	13354467	<p>本门课程，让我体会到了web设计的改进过程。首先，好的编程工具能让我们从繁琐的项目管理、代码编译等工作中解脱出来。另外，如livescript等近乎自然语言的代码，使得代码变得通俗易懂。而“自顶向下”的编程方式能够让思维更加清晰。对于TOP，是广度优先的过程，细节的实现只需留给下个阶段的工作，这样能让我们有一个清晰明确的思路，每个时刻都能有全局观。</p> <p>在小组作业中，由于想法不完善，我们在开始阶段经历了很多挫折。后来我们重新讨论了我们要做什么，然后想着怎么去实现，最后付诸于行动。这一过程也让我们体会了如何一步步建立起自己的项目。</p>
杨振杰	13331311	<p>这个学期的软改学习，虽然很辛苦，但是收获颇丰。一开始对javascript的学习，使我能够更深入地了解javascript的思想，那就是“一切皆对象”。在此之上，通过prototype和constructor就能够实现类的继承与派生。虽然对原型链的理解并不是十分深刻，但是相比与之前对javascript的肤浅认识，已经好上许多了。每种语言，都有自己的奥妙，不能简单主观地轻视某些语言。然后在javascript的基础上又学习了livescript。虽说livescript的本质仍是javascript，但是也有自己的深奥之处。经过一段时间的学习，虽然没有完全掌握，但是我认为livescript就是语言发展的趋势，语言必定是越来越简单，越来越接近自然语言，程序员不再需要被各种枯燥的语法束缚。在此之上又学习了expressJS和meteor这两种框架，第一次知道什么是“响应式页面”“单页面设计”，这应该也是今后网页发展的趋势。期末项目的开发，使我明白一个项目只有立意正确才能够使项目走的更长远，在立意正确的基础上，架构、框架、团队协作则决定了开发的效率。以上所学习的知识，不能算精通，但是触类旁通，对以后的学习和工作也是一笔重要的财富。</p>
牛仁鹏	13331203	<p>经过一学期的学习，我见识了许多web的新技术，同时也理解了一些TOP思想。总的来说一学期的课还是挺紧的，每星期都有新的工具去学习。但是我觉得通过一学期的学习是不够的，我仅仅做到了了解这些工具，并不能说是掌握了，所以我打算利用暑期的这些时间去彻底学习一下，毕竟这些工具可以极大地缩减web开发所花费的时间。总的来说，这学期的收获还是蛮大的。</p>

姓名	学号	心得总结
苏永健	13331227	<p>对王青老师的软件过程改进这门课,感触最深的就是其中的 TOP 思想了。</p> <p>第一次接触到这种神奇的编程思想的我,在看老师课上的 Live coding show 时,简直是惊呆了,原来真的是可以将自己的想法转换成可运行的代码的啊,虽说当时不是很了解 livescript 的语法,各语句理解起来有点乱,但起码能知道编写程序的思路非常清晰而又直观的体现在了函数的命名上。结合老师展现给我们的 TOP 思想,在了解了 liveScript 的基本语法后,我开始了基于 liveScript 的 thinking-orient programming 编程之旅,收获良多。</p> <p>还有就是在后期的项目改进部分课程里也学到了很多东西。虽然 ideas 不断被质疑和否决,但我们从错误中吸取了很多经验和教训。</p>
王文茏	13331256	<p>这个课程的前半部分从css和JavaScript的常用的重点和一些较难理解的细节部分讲解,逐步过渡到一些编程工具和框架的介绍和使用入门,其中包括node.js, grunt, livescript, jade等等。最后介绍了一个可以大大减少代码量,提高迭代速度的框架meteor。总的来说这部分知识,特别是代码能力方面学到的东西还是很多的,但作业方面我觉得还可以再斟酌一下。一方面一次作业的代码量会比较大,其中就有两三次作业延期的情况,而且这样ta批改的压力也会比较大,最后一次作业貌似现在分数都没有出来。。。</p> <p>第二部分的课程是模仿真实开发环境来快速开发一个网站,以期让同学们体会到正式入行后可能面临的问题,以及如何把一个想法变成一个真正的产品的方法和工具。这对于IT从业人员其实也是挺重要的,但我觉得这部分可能对于有志于创业的人更重要,平常就业的话通常会从改善现有产品开始,所以我觉得也许可以该课程所有小组都开发同一款产品,或者改进同一款产品的不同功能,然后下一学年的课程的各个小组再在上一学年小组的产品的基础上再做改进或重构,这样一年年下去,产品一次次迭代。这是我从邹欣的讲解软件工程的《构建之法》这本书里看到的,觉得是种不错的学习方式。</p>
陈纪云	12330018	<p>这个学期,学习了很多框架知识,特别是到NodeJs的部分。纯英文的网站看起来很痛苦,但是最后的结果却是很好。说实话,我挺喜欢这门课前面的部分,每个星期一个Project,然后熬夜很痛苦的把它写完,但是每次都收获了很多,很开心。我想以后都会记得曾经有过这么一个拼命学习的时光的,谢谢老师!</p>