# 引言

近年来，互联网的普及以及在线协作创新的发展使得开源软件的研究和开发再次成为人们关注的热点。从广泛使用的Linux操作系统，到市场占有率第一的Android手机操作系统，再到热点大数据使用的Hadoop分布式系统架构，这些都是为我们现代化生活做出巨大贡献的开源软件。开源软件具有成本节约、高效稳定等特点，不仅普通用户喜爱此种模式，商业软件公司和新兴互联网公司也纷纷开始尝试。2014年底，微软开源了.net框架；2015年初google开源了c++版的mapreduce。从这些互联网巨头的动作中，开源软件的受欢迎程度以及影响力可见一斑。

开源软件是源码可以被公众使用的软件，并且软件的使用、修改和分发也不受许可证的限制。开源软件的开发工作是可以有不同地理位置、不同时区、不同肤色、不同文化背景的人一起通过互联网完成的。Raymond[19]认为，与传统商业软件的开发相比，开源软件的开发就像是一个自由市场。开源软件的开发者、使用者都可以提出自己的建议。成功的开源软件用户满意度十分高，有着非常高的效率、稳定性和快速的问题解决能力。正因为开源软件的这些商业软件不能比拟的优点，越来越多的个人和公司被它吸引。国内的华为、阿里、联想等，在开源项目中投入了大量的资源，国外的google、ibm、英特尔等也热衷于开源研究。

早在1991年开始的Linux内核项目就是至今最著名、最成功的开源团队开发的项目之一。[6]芬兰的Linus Torvalds尝试在英特尔x86架构上开发的类似Unix的自由免费的Linux内核虽然只是完整的Linux系统中比较小的一部分内容，但是这部分内容决定了系统如何工作，正如其名，是Linux中不可或缺的一部分。[7]

在开发者的内部版本管理上，Linux内核采用Git版本控制系统进行版本与代码管理。[6]Git作为一个版本控制系统，它的优势主要体现在其“去中心化”的设计思路，每个开发者的版本库都是自己的中心版本。同时，Git具备机器优秀的分支创建、操作以及删除的能力，衍生出了非常多的基于分支的高效的协同工作流模式。

Git的使用，使Linux内核的开发更具有效率，不同地域的工作人员可以自行安排开发时间，在本地开发、测试完成之后上传至自己的分支，再与主项目进行合并，这样就能实现不同时间开发人员的同步开发。

为了降低入门开发者的学习成本、提高开源软件的开发效率、增强用户体验开源软件的便利性，我们提出本文所引述的平台思想。

本平台是在git的开源开发模式基础上使用scrum的敏捷开发思想、由用户组成团队完成开发、再向对其有兴趣的用户进行辨认、推送，以此来达到共同完成、共同分享的目标的一个众包式开发社区。

相关研究部分简单地介绍了平台的思想和一个常规的业务过程。平台的思想来源于许多现存的开源众包形式平台，文章分析现有平台的优势与劣势，体现本平台思想的特点。然后通过一个常规业务过程向用户展示平台的可行性以及与其他平台的不同。

为了优化推荐，提高用户找到自己感兴趣的开源软件的便利性，平台建立了基于开发过程的成熟度评分机制，在相关研究之后会重点解释成熟度的含义以及成熟度的计算。成熟度为接下来的排序、推送过程提供了强有力的依据。

在演绎成熟度评分机制之后，即是针对用户兴趣的推送机制的介绍。推送机制有两种，一种是平台特色的成熟度推荐，一种是常用的热度推荐。除此之外，还有其他二维排序机制配合推送，这些都将在最后一部分进行详细介绍。

# 相关研究

“众包”这一概念从提出至今得到了工业界和学术界的广泛关注。众包(crowdsourcing)此概念是美国杂志《连线》的记者杰夫·豪(Jeff Howe)在2006年5月提出的。他对“众包”的定义是：“一个公司或机构把过去由员工执行的工作任务，以自由自愿的形式外包给非特定的（而且通常是大型的）大众网络的做法．众包的任务通常由个人来承担，但如果涉及到需要多人协作完成的任务，也有可能以依靠开源的个体生产的形式出现．”[3]

相似的另一个概念“外包”是指用合同的而形式把任务派发给另外的个人或组织来完成。外包业务一般是借助计算机来完成的任务，比如软硬件的开发和维护。软件提供方通过整合硬件、程序来完成外包业务，比如提供软件服务、外包数据库、制作网站等。但是有一些任务不是算法能够解决的，这类“机器难问题”就可以通过众包的形式来解决，将任务分发给互联网上未知的大众来解决，比如wikipedia、recaptcha等。现实生活中有许多类似场景，这也就意味着众包思想广阔的发展空间和光明的前景。

Recaptcha[4]是由卡内基梅隆大学所发展的系统，主要目的是通过captcha[5]技术来帮助典籍数字化的进行，这个项目将电脑扫描下来且无法识别的文字显示在captcha问题中，通过人类在完成captcha问题的同时进行识别。

Github的用户间协作完成项目亦可近似地认为是众包的形式。一个用户提出相关需求并做出初步实现，然后想其他用户展示产品，其他用户帮助其进行更进一步的完善。但是在众包更加广泛应用的现在，用户群体开始变大，不再仅仅局限于拥有编程能力或是解决能力的开发者，用户也可以是提出自己需求、展示市场痛点的策划人。这些用户（下文简称“策划者”）没有一定的开发能力，无法通过初步的模型结构来给其他用户展示自己的想法，只能通过自然语言来表现自己的想法，本文所构思的平台就是给予用户这个功能，让没有开发能力的用户也能成为“开发者”，以此达到拓展用户群体的目的。

与其说是拓展用户，开拓新用户群体更像是帮助策划者让他们能够通过众包的形式来实现自己的想法，或是通过与其他更加了解此领域的用户的交流发现他的想法其实是天马行空、并不现实的。

基于这类用户以及具有开发能力的开发者的需求，平台应该提供给所有用户以文字描述自己需求、能够方便地浏览高质量内容、能够高效地加入一个开源开发团队等的功能。

当策划者有一个灵感的时候—有时仅仅是一点小小的需求，就可以直接在平台上提出此创意，然后系统会由标题抓取关键词，并首先自动向策划者提供相似的创意发展出来的项目。如果策划者觉得系统推荐的以往的创意项目符合自己的预想，那么可以自行关闭创意；如若觉得并不符合自己的需求，可以再添加更加具体的描述，然后发布此创意。接下来开始进入第一阶段，平台用户一起讨论可行性以及再进一步的润色创意。如果在此阶段，经过讨论发现此创意实现难度极大甚至基本无法实现，策划者可以自行关闭创意。如果经过润色补充后，创意逐渐成为一个充实、可执行的项目，那么策划者可以广招贤士，开始进入第二阶段—组建项目团队阶段。

组建团队不仅是完成项目的必要的环节，也是增加社区活力，贯彻众包思想的必须阶段。研究表明，在线交流与社群关系的维护是用户参与团体行为的基础[15]。众包的成功依赖于社区用户的群体智慧，需要社群成员间的协作以及用户与社区间的良性互动。[16]

在组建项目团队阶段，策划者需要首先选择一个项目负责人，此负责人在此后的进度中拥有和策划者相当的权限，考虑到负责人对项目开发的流程了解程度一般大于或等于策划者，平台给予负责人个别操作的权限甚至可以超过策划者，使负责人作为此项目的平台管理之外的最高权限角色之一。在确定项目负责人后，平台会开放平台全用户对此创意开发团队的申请权限，所有的用户都可以向开发团队提出申请加入开发团队。同时，系统会自动地向关注此创意的用户发出邀请，收到邀请的用户也可以通过邀请进入此创意，然后选择是否加入开发团队。

申请阶段会持续一段时间（1d，3d，5d，7d），同时，项目负责人开始审核用户发来的申请，在申请中会有用户的自我描述，建议此时尽量展示自己的优势以及对项目有帮助的方面，增加自己申请通过的几率。在项目人员基本确定后，项目开始正式进入开发阶段。如果负责人认为项目开发人员并不够，可以再延长最后一期申请时间，也可以直接裁定团队建设失败，做好记录后关闭项目开发。

进入正式开发阶段后，需要选择一种软件开发模型给团队使用，平台推荐使用scrum敏捷开发过程模型，并为此提供了充分的作业空间。瀑布模型、快速原型模型、增量模型、螺旋模型、喷泉模型灵活性不高，在本平台上的项目需求和人员的变动相对较大，不适合使用这些开发模型；而rational过程和微软过程虽然比较全面，但是相对工作比较繁琐。Scrum方法是目前全球最流行与最有效的敏捷项目管理理念与方法之一，是一种快速增量交付软件产品的模型，在构建产品过程中创建产品的内部团队并与其他用户高度协作。[8]这些特点都使得平台上的项目更适合scrum开发模型。当然，用户也可以不使用scrum敏捷开发过程模型，充分使用平台提供的功能空间进行开发也是允许的。

在正式开发阶段，项目组需要先正式地第一次确定需求，分配工作并进行记录。讨论阶段，平台提供项目组专属的讨论组，并提供一定的存储空间存放记录的项目文档，需要注意的是，这些文档意味着正式的记录，是可以被平台上的每一个用户浏览的。

正如“基于 Scrum 的知识共享模型研究[2]”中提到的，在开发阶段会有计划会议、每日会议、评审会议、回顾会议等会议。在这些会议中，讨论部分都可以在讨论组中进行，而重要的决策需要记录员写入文档后存储在此项目的文件库中，此文件库如上文所说是可以被全体用户查看的，也是这个创意项目共享的关键。

陆丹(2013)在对“众包”特征分析的基础上，提出众包面临四种风险：能力风险、组织管理风险、知识产权风险和信息风险，并论述每种风险产生的原因，提出风险规避的解决办法，但具体实施的可操作性需要进一步 完善。[17][18]

具体到本平台，为了规避商业化众包的风险，同时作为一个共享想法、协同实现的开放式平台，idea的内容采用了cc by-sa 3.0 的思想，将想法的版权也共享开。意思是作者允许任何人在限定条件下，不必经作者同意而复制或传 播其作品。“CC”是英文短语“Creative Commons”的缩写， 2006年3月正式确定其中文译名为“知识共享”，使用这个协议的作品，除特殊说明外，允许他人免费拷贝、分发(任何 形式)、讲授、表演。但这并不意味作者放弃了自己的著作权， 而是指在特定的条件下将其部分权利授予给公共领域内的使 用者。[14]但是共享并不意味着人人都可修改。每一次对文件库的修改操作只能由项目组人员进行。在项目结束后，项目组也将被收回对文件库的修改权限。

# 项目成熟度机制

由于项目多种多样，项目质量的好坏参差不齐，有必要制定一个衡量项目优劣的机制来控制项目的质量。比如赵岩露等[13]建立微博用户兴趣模型,并提出基于微博用户兴趣模型 的发现算法。

而我们提出了一个体现项目当前实现状态的指数——成熟度，用数字直观地表现一个项目在实现过程中获取的实际价值。之后将要介绍的项目成熟度机制就是依靠成熟度对项目进行筛选排序，让更多优质的项目更有机会展现它的价值。理想的成熟度计算是通过系统智能地抓取相关信息进行判断给分的，但是在平台早期因为开发资源地约束，可以通过其他更简单手段进行可以节约平台的开发资源，具体在每一个环节将详细介绍。

在此平台，成熟度机制主要是为了帮助用户发现精品内容，为了客观地给出项目的成熟度指数，平台采用公式进行计算，以下三方面均是重要考量因素：

1. **项目组成熟度**

项目组成熟度是指项目组人员质量、数量、分工结构的完善程度。

人是生产力的第一要素，在项目中，开发、管理都是通过人完成的。企业人员结构是否合理，直接影响到企业的运作效率。[10]同理，项目人员结构的合理程度，是影响项目开发效率的重要影响因素之一。一个专业的用户只有在担任了他所擅长的岗位之后才能最大化地发挥他的能力。

在平台前期，可以先使用项目组填写的方式，这种方式相对来说更考验平台用户的诚实度，解决方案是可以增加对此情况的举报功能，举报成功后将扣除一定分数以示惩罚。

为了计算项目组参与人员的质量，平台推出等级制度，用户在平台上参与的活动获取积分，根据积分获得等级。同时加入分类标签制度，将用户所擅长的技能分门别类，作为用户的个人标签，在项目分工时也更方便，其他用户亦可对这些标签进行证明。若在CSDN、Linkedin、Github等较为出名的网站上有较高的知名度，平台可以直接赋予其一定的积分和高等级标签。

项目组成熟度的计算就是成员等级的总和。

1. **项目需求成熟度**

项目的需求表现了项目的细节完成度，一个细节考虑越多的项目往往意味着更加成熟的项目。

在平台开发初期，系统不够智能，对于需求成熟度可以通过项目组填写的需求数量，以及用户通过点赞对某些需求的评分来综合计算。一般来说，需求越多的项目在此环节得分越多，但是为了减少不重要需求对此环节的影响，项目需求成熟度得分与项目需求数量的关系应该是a小于1大于0的幂函数。（配图）

y

x

y=x^a(0<a<1)

1. **项目关注度**

一个内容丰富、创意新颖的精品项目一般来说拥有比其他项目更多的关注，所以从用户关注数量、浏览数量上可以计算项目关注度的评分。此影响因素存在逻辑上的正反馈死循环，如图所示（更多的用户关注→关注度评分更高→项目成熟度评分更高→更容易有更多的用户来关注此项目→。。。）。

但是一个项目不可能长时间保持高关注度，所以在此环节的评分计算上，要根据时间对关注度打折，算法公式如下：

关注度=((关注数)^(1/2) + (浏览数)^(1/3)) \* (1/(项目发布至今的天数/7+1))

# 基于成熟度评分的管理过程

众包的工作方式可以帮助任务需求这获得大量自由工作者，通过这些工作者的智慧解决实际问题，然而，有的众包工作者为了使自己的利益最大化，没有认真地为任务需求者工作，所提交的结果质量低下，违背了需求者发布任务的初衷，甚至产生了适得其反的结果，识别出欺骗类型的工作者是一项具有挑战的工作。[9]

平台对于项目开发人员中的欺骗者的识别是通过项目的负责人进行的，项目负责人在招收人员、结构调整、项目开发的过程中都可以使用自己的办法来识别这些工作效率低的项目人员并进行处理。对于项目中的低质量项目，平台是通过上文介绍的成熟度评分进行识别的。通过平台的成熟度评分计算系统可以计算出项目的具体得分，在之后的管理过程中，可以使用成熟度评分作为依据进行管理。

当用户有浏览平台推荐的精品内容的需求时（项目广场或者其他名字），平台将会按照项目的成熟度评分进行排序，为用户展示最有价值的项目。在基于项目成熟度的排序过程中，平台还提供分类查看的功能，以项目建立时贴的类别标签作为分类依据，分类进行排序，然后再向用户展示内容。

再基于项目成熟度评分的排序过程之外，平台还提供按照热门程度（浏览量、评论量、关注量）为依据的排序过程。在基于热门程度的排序过程中，同样支持分类查看。

# 引用

1. GitHub:全民一起“打怪升级”
2. 使用Github实现高效的团队协作开发
3. 众包技术研究综述
4. <https://zh.wikipedia.org/wiki/ReCAPTCHA>
5. <https://zh.wikipedia.org/wiki/captcha>
6. Lillux内核开发者协作模式研究
7. KroahHartman G， Corbet J。 McPherson A． Linux kernel development[J]． the Linux foundat ion． 2008．
8. Scrum方法的研究与分析
9. 众包质量控制策略及评估算法研究
10. 谈企业人员结构
11. 赵岩露,王 晶,沈奇威. 基于特征分析的微博用户兴趣发现 算法[Z]. 东信北邮信息技术有限公司专栏, 2012, (11):79- 83
12. 从"保留所有权利"到"保留部分权利"——解析"知识共享组织"及"CC"协议
13. ] Xiao-Liang Shen, Matthew K O Lee，Christy M K Cheung. Ex⁃ ploring online social behavior in crowdsourcing communities: a relationship management perspective[J]. Computers in Hu⁃ man Behavior, 2014, 40(1):144-151.
14. 众包社区用户持续使用行为研究\* ——基于ECM-ISC和承诺信任理论
15. 陆丹，徐国虎．基于“众包”的企业创新模式研究 [J]．物流科技，2013，8：127—129．
16. 陆丹．互联网环境下众包风险的识别与规避[J]． 物流工程与管理，2013，35(4)：118．120
17. Raymond E．The cathedral and the bazaar[J]．Knowledge，Technology＆Pol icy， 1999，12(3)：23—49．