# 引言

从开始到现在，计算机系统已经经历过了四个发展阶段，然而人们仍然没能完全摆脱“软件危机”的困扰，软件已经成了限制计算机系统继续发展的瓶颈。

为了更加有效地开发和维护软件，软件工作者们在20世纪60年代后开始努力研究消除软件危机的方法，后来逐渐形成了一门学科—软件工程。

“概括地说，软件工程是知道计算机软件开发和维护的一门工程学科。采用工程的概念、原理、技术和方法来开发与维护软件，吧经过实践考研而证明正确的管理技术和当前能够得到的最好的技术方法结合起来，以经济地开发出高质量的软件并有效地维护它，这就是软件工程。”【1】

为了消除“软件危机”带来的对软件正常使用的威胁，我们应该以软件工程的思想开发和维护我们的软件，特别是对于我们软件工程专业的学生。但是在日常学习、开发过程中，我们发现非大型it企业的软件从业者旺旺没有良好的软件工程开发思想，只注重代码不注重文档，只为了实现功能而不为了高效地实现功能。我们发现市场上协作平台不少，但是对于使用软件工程思想、并为整个开发过程提供直观、简洁、高效的社区却不多，甚至可以说是一片空白。而本文所介绍的社区—致力于帮助初级软件学习者、软件从业者更好地学习软件工程思想，正是要填补此类市场空缺。

软件开发过程可以概括地说是为了完成客户需要的软件，什么人（who）、在什么时候（when）、做什么事（what）、和怎样（how）做这些工作用来实现目标。

为了高效地、简洁地描述软件过程，我们一般使用生命周期模型。生命周期模型包含了生命周期的各个阶段以及每个阶段的执行顺序，也称作过程模型。

生命周期模型有很多，我们常用的生命周期模型有八种，包括瀑布模型、快速原型模型、增量模型、螺旋模型、喷泉模型、rational统一过程、敏捷过程与极限编程以及微软过程。

使用传统的瀑布模型开发软件，有以下特点：阶段间具有顺序性和依赖性、推迟实现的观点、质量保证的观点。可是传统的瀑布模型是非常理想化的，实际上，我们在工作过程中不可能不犯错误。实际的瀑布模型通过反馈环来解决文档和设计中的缺陷和错误。所以瀑布模型有许多优点，可以强迫开发人员采用规范化的方法进行开发，严格规定了每一个开发阶段需要编写提交的开发文档，每一个阶段交出的产品都必须经过质量保证小组的仔细测试。正因为此，瀑布模型的缺点也很明显，它是由文档驱动的。用户只能通过文档了解产品是什么样的。但是这样静态、抽象得描述并不能具体地向用户展示设计者的想法。而且用户的需求并不是不变的，在设计过程中，很可能会进行一些调整，是的最初提出的需求就会变得不是那么的适用了。这就很可能导致最终做出来的产品和用户实际想要的并不是一回事，不能真正地满足用户的需求。

为了使用户的真实需要得到满足，我们又建立了快速原型模型。所谓快速原型是快速地建立起的可以运行的程序，这个程序所能完成的功能一般是最后成品的所有功能的一个子集。快速原型模型是不带反馈环的，因为原型系统就已经与用户交互后得到了验证，根据这个产生的规格说明文档是正确地描述了用户的需求。由于原型系统经常改变，可能会导致一些不必要的缺点，如果此时不解决，那么合并起来时也会以低效状态展示给用户。所以如果采用快速方法开发，系统的稳定性和优化性也会相对的没那么好。

增量模型也称作渐增模型。采用增量模型时，我们是把整个软件项目当成一系列的增量组件来设计、编码、集成和测试的。每一个组件由相互的几个模块构成，而且能够完成一定的功能。使用增量模型时，先建立一个核心模型，提供最核心的功能，然后在此基础上添加其他的功能模块。使用瀑布原型和快速原型开发模型时，我们的目标都是一次性地满足所有需求，然后一次提交给客户。而增量模型不同，甚至正与他们相反。增量模型时分批次地、逐步向客户提交产品。能在短时间内向用户提交一个可以完成部分功能的产品，是增量模型的优点之一。使用增量模型的缺点在于，当添加一个新的增量组件到现有的软件结构时，必须不破坏原有的软件结构。所以整个软件体系结构应该是开放的、灵活的，这样对于前期设计阶段的要求就会高于其他模型。相对的，开放式的软件结构也为后期的维护工作带来了便利，比其他模型更加方便维护。

软件开发要面临许多风险，例如，成品交付后客户不满意，或者到了预定的时间还没有完成开发，实际开发成本比预算要多等。软件风险是每个软件项目不得不面对的困难，越大越复杂的项目意味着更多更大的项目风险。所以，在软件开发过程中，应该及时的识别和分析风险，并且采用适合的措施来减少或消除风险的危害。

构造原型是使一些风险降至最低的方法之一。螺旋模型的基本思想就是通过不断地制造原型来降低风险，我们可以把它看作是每个阶段前都添加了风险分析过程的快速原型模型。螺旋模型强调对可选方案和约束条件的强调，有益于对现有软件的重用，也有益于吧软件质量作为软件开发过程中的一个目标。螺旋模型减少了过多的测试或者是测试不足所会带来的风险。最重要的是，螺旋模型中的维护只是模型之中的一个周期，维护与开发之间并没有本质上的区别。螺旋模型的优势在于，它是风险驱动，当然，这也是它的一个劣势。如果开发人员没有丰富的风险评估经验和这些方面的专业知识，那么有可能会出现当整个项目走入死胡同时，开发人员还没有发现。

迭代是软件开发过程中普遍存在的一种属性。一般来说，面向对象思想的软件开发过程的首要工作是建立业务相关的对象，并在接下来的设计与编码中逐渐充实对象的属性和功能。由于使用了同一概念“对象”，那么在分析、设计、实现阶段的区分也就不明显、不重要了。各个阶段使用统一的概念，对于整个开发过程是一致的，甚至是无缝衔接的，那么就很容易实现反复迭代，不断地充实、深化认识。所谓的喷泉模型就是典型的面向对象的软件开发过程模型之一。喷泉模型是最接近方法论本质的模型方法，但是有时最难用教条的工程学来表述和管理。

Rational 统一过程是有rational软件公司推出的一种完整且比较完美的软件开发过程模型。RUP（Rational unified process）总结了多年来商业化验证了的最有效的6条开发经验，我们称之为“最佳实践”。最佳实践包括—迭代开发、管理需求、使用基于构件你的体系结构、可视化建模、验证软件质量、控制软件变更。Rup强调使用渐增和迭代的方式来开发软件，整个软件开发过程由多个迭代过程组成。每次迭代中只完成部分需求，并针对这些需求进行分析、设计、实现、测试和部署等工作，每一次的迭代都是在已完成的项目的基础上进行的，每次给软件添加一些新功能，照这样循环往复地进行下去，直到完成整个项目。

为了使开发团队具备高效工作和快速响应变更的能力，2001年2月17名著名的软件专家联合起草了敏捷软件开发宣言。敏捷软件开发宣言由四个价值观声明组成：“个人和交互胜过过程和工具；可以工作的软件胜过面面俱到的文档；客户合作胜过合同谈判；响应变化胜过遵循计划。”根据上述价值观提出的软件开发过程称为敏捷过程，敏捷过程中最重要的是极限编程。以极限编程为代表的敏捷开发过程具有对变化和不确定因素响应迅速的特点，能够更好地适应商业竞争环境下小型项目对资源和时间的约束。

作为世界上最大、也几乎是最成功的软件公司之一，微软也具有自己的一套开发模型。微软开发过程遵循以下准则：“项目计划应该兼顾未来的不确定因素；用有效的风险管理来减少不确定因素的影响；经常生成并快速地测试软件的过渡版本，从而提高产品的稳定性和可预测性；采用快速循环、递进的开发过程；用创造性的工作来平衡产品特性和产品成本；项目进度表一个具有较高稳定性和权威性；使用小型项目组并发地完成开发工作；在项目早期吧软件配置项基线化，项目后期则冻结产品；使用原型验证概念，对项目进行早期论证；把零缺陷作为追求的目标；里程碑评审会的目的是改进工作，切记互相指责。”微软模型作为另一种适用于商业环境下对项目有有限资源和有限时间约束的软件过程模式，微软过程集合了rational过程和敏捷过程的许多优点，是对众多成功项目开发项目经验的正确总结。另一方面，微软过程也有自己的缺陷，比如对方法、工具、产品的论述没有rup和敏捷过程全面等。

我们的项目提倡使用软件工程的思想，使用适合的软件开发过程进行我们生活中一些想法的共同开发。由于人员的不稳定性，需求的不稳定性，我们需要的是一个灵活多变的软件开发过程模型来减少人员、需求的变更对项目的影响。考虑到我们日常创意的需求并不多，项目比较小，项目人员流动性大，积极性可能会不高，我们更加建议使用敏捷开发过程思想来进行开发，并且我们的项目也为敏捷过程思想提供了工作的空间和舞台。

# 总结文献

如今是一个大数据时代，也将是一个人人都有创意的时代，“大数据的出现对创意社会的来临起到了催生作用”。

[1]谢俊贵.信息社会之变：大数据催生创意社会?[J].广东社会科学,2016,(5):185-195.DOI:10.3969/j.issn.1000-114X.2016.05.020.

[2]田少煦,孙海峰.创意设计的发展走向与核心竞争力[J].深圳大学学报（人文社会科学版）,2010,27(3):136-141.DOI:10.3969/j.issn.1000-260X.2010.03.023.

[3]殷茗,蒋丹,马静等.异地分布式敏捷开发团队协作合作博弈研究[J].西北工业大学学报,2016,34(4):713-718.DOI:10.3969/j.issn.1000-2758.2016.04.025.

[4]李文浩.基于ASP．NET MVC模式和Scrum敏捷开发的服务外包平台[J].有色金属加工,2016,45(5):60-63.DOI:10.3969/j.issn.1671-6795.2016.05.015.

[5]戴洪蕾,王建国.基于敏捷技术的敏捷开发辨析与应用研究[J].中国市场,2015,(21):53-56.DOI:10.13939/j.cnki.zgsc.2015.21.053.

Scrum方法的灵活性高，在人员流动性大、项目需求变更频繁的情况下，“毫无疑问，Serum已经是当今最流行的敏捷项目管理框架与实践。”

[6]张智海,周国祥.Scrum方法的研究与分析[J].合肥工业大学学报(自然科学版),2010,33(2):197-200.DOI:10.3969/j.issn.1003-5060.2010.02.008.

[7]蒋翠清,赵小英,梁昌勇等.基于Scrum的知识共享模型研究[J].情报杂志,2011,30(3):101-105.DOI:10.3969/j.issn.1002-1965.2011.03.021.

[8]徐欣欣,罗晓斌.浅谈敏捷软件开发中Scrum方法的运用[J].计算机光盘软件与应用,2013,(7):266-267.

[9]邓爱林,朱扬勇,施伯乐等.基于项目评分预测的协同过滤推荐算法[J].软件学报,2003,14(9):1621-1628.

# 系统实现

我们在日常生活中常会有各种奇思妙想，在这个信息化的时代，在这创意化的时代，每一个想法或许都会有极大的价值，我们idea的主题思想就是帮助我们这些想法以一个可行性高的方式、搭配一个灵活的志愿团队、用最高效简洁地过程来论证、实现这些想法。

当想法提出者有一个灵感的时候—仅仅是一点小小的需求，此时可以直接向系统提出此创意，此时系统会由标题抓取关键词，并首先自动向提出者提供相似的创意。如果提出者觉得系统推荐的以往的创意符合自己的预想，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，然后发布创意。此时开始进入第一阶段，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，提出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，那么提出者可以广招贤士，开始进入第二阶段—组建项目团队阶段。

在组建项目团队阶段，提出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提出者相当的权限，个别操作可以超过提出者作为此项目的平台管理之外的最高权限。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。

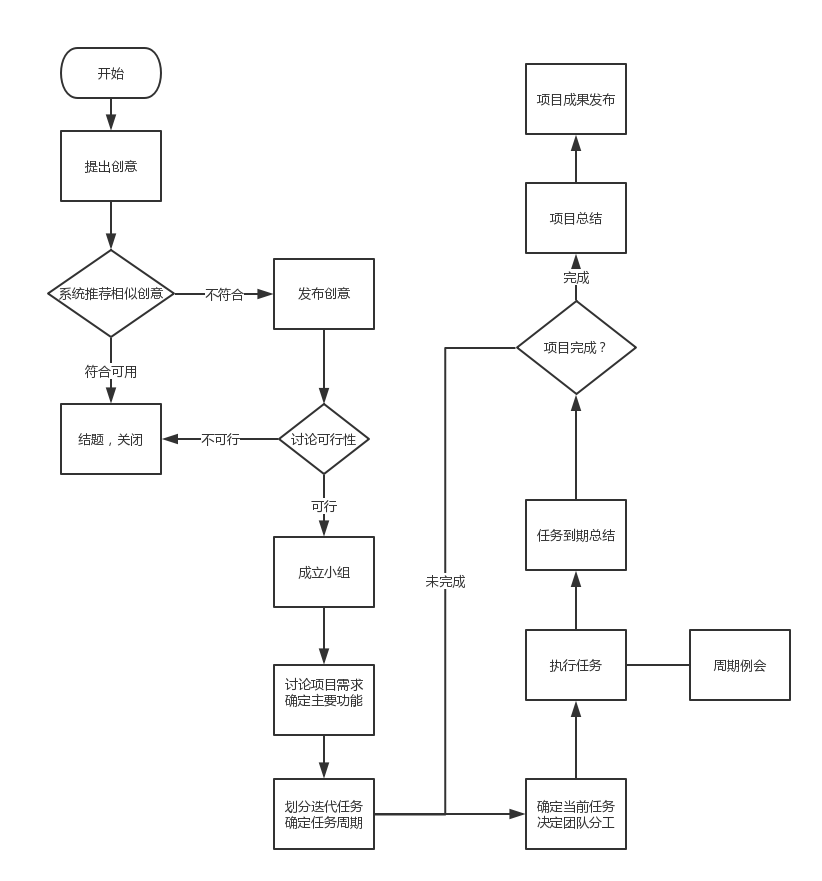
在一段时间（1d，3d，5d，7d）的申请阶段同时，项目负责人开始审核用户发来的申请，在申请中会有用户的自我描述，建议此时尽量展示自己的优势以及对项目有帮助的方面，增加自己申请通过的几率。在项目人员基本确定后，项目开始正式进入开发阶段。如果负责人认为项目开发人员并不够，可以再延长最后一期申请时间，也可以直接裁定开发失败，做好记录后关闭开发。

进入正式开发阶段后，正如上文所论述的，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，所以平台推荐使用scrum敏捷开发过程模型，并为此提供了充分的作业空间。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，也是这个创意共享的关键。

作为一个共享想法、协同实现的平台，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。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。



服务器、客户端实现环境。

主要业务逻辑场景。

主要业务逻辑实现。

系统模块结构图与介绍。

系统数据流图与介绍。

# 引用

1. 第一章 1.2.1 软件工程的介绍。软件工程导论（第六版），张海藩 牟永敏 编著，清华大学出版社。
2. 基于 Scrum 的知识共享模型研究\*  
   蒋翠清 赵小英 梁昌勇 丁 勇  
   ( 合肥工业大学管理学院 合肥 230009;  
   过程优化与智能决策教育部重点实验室 合肥 230009)