

△
Addison
Wesley

计算机技术
译林
精选系列

小组软件 开发过程

[美] Watts S. Humphrey 著
袁昱 译

人民邮电出版社
www.pptph.com.cn

计算机技术译林精选系列

小组软件开发过程

[美] Watts S. Humphrey 著

袁 昱 译

人民邮电出版社

图书在版编目 (CIP) 数据

小组软件开发过程/ (美) 汉弗莱 (Humphrey.S.W) 著; 袁昱译. —北京: 人民邮电出版社, 2000.10

(计算机技术译林精选系列)

ISBN 7-115-08763-6

I. 小... II. ①汉...②袁... III. 软件开发 IV. IP311.52

中国版本图书馆 CIP 数据核字 (2000) 第 46333 号

计算机技术译林精选系列

小组软件开发过程

◆ 著 [美] Watts S. Humphrey

译 袁 昱

责任编辑 俞 彬

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@pptph.com.cn

网址 <http://www.pptph.com.cn>

北京汉魂图文设计有限公司制作

北京顺义振华印刷厂印刷

新华书店总店北京发行所经销

◆ 开本: 787 × 1092 1/16

印张: 17.75

字数: 421 千字 2000 年 11 月第 1 版

印数: 1 - 5 000 册 2000 年 11 月北京第 1 次印刷

著作权合同登记 图字: 01 - 1999 - 2563 号

ISBN 7-115-08763-6/TP·1809

定价: 34.00 元

《小组软件开发过程》(简称“TSPi”),是美国 Embry-Riddle Aeronautical 大学为计算机科学系研究生和高年级本科生开设的一门软件工程课的教科书。这本书系统地论述了如何以开发小组的形式来进行软件的开发,并对开发过程作出了具体而详尽的指导,包括小组成员之间的协调、进度的管理、质量的控制等等令读者最感兴趣的方面。

全书共四个部分:第一部分——绪论,包括前两章,是对理论的简单介绍,介绍了什么是 TSPi、TSPi 的组织结构等内容。第二部分——TSPi 过程,包括第三章到第十章,则是整个小组研究周期的详细内容,详细解释了小组软件开发的步骤,并且给出了 TSPi 完整形式的例子。第三部分——小组角色,包括第十一章到第十五章,提供了小组成员角色的细致描述:小组领导者、开发经理、计划经理、质量/生产经理,以及技术支持经理。第四部分——使用 TSPi,包括第十六章到第十八章,讲述了在使用本书的过程中需要注意的一些原则。全书内容难度适中,全面生动地阐述了软件工程的基本知识。

本书实用性与可读性较强,适用于软件开发项目经理、程序员和一般编程爱好者在开发软件时参考,也可作为高等学校计算机软件工程课程的参考教材使用。

版权声明

Watts S. Humphrey: Introduction to the Team Software ProcessSM

Authorized translation from English language edition published by Addison Wesley Longman, Inc.

Copyright © 2000 by Addison Wesley Longman, Inc.

All rights reserved. For sale in Mainland China only.

本书中文简体字版由 Addison Wesley 出版公司授权人民邮电出版社出版，未经出版者书面许可，对书的任何部分不得以任何方式复制或抄袭。

版权所有，侵权必究。

信息时代的大潮，冲击着世界的每一个角落。计算机技术的应用，正在越来越多地渗透进人们生活的方方面面。谈起计算机技术的飞速发展，人们通常最容易想到的是硬件制造水平的进步，例如 CPU 主频的提高，3D 显卡性能的飞跃等等。然而，软件业的发展速度也是相当惊人的。软件的编写过程，早已不仅是个把程序员的埋头苦干，而成为了一项现代化大规模生产的产业。微软的 Windows 98 和 Windows 2000 源程序多达几亿行。这样大规模的程序，如果没有一套科学的开发机制，不要说提供给用户使用，就是让它“Run”起来，也是不可能的。

我们欣喜地看到，我国的软件产业，也正在蓬勃发展起来，在软件的本地化、民族化方面取得了长足的进步。在其他领域也有了可喜的成就，金山词霸、WPS、超级解霸等等，都是成功的例子。另一方面，中国人在软件方面的才能，也得到了世界的承认，各大软件公司都活跃着中国人的身影。但是，仔细分析国内软件产业的现状，我们不能不承认这样一个事实：那就是，就国内大多数中小软件公司乃至部分大公司而言，软件的开发还处在“手工作坊”阶段，开发过程缺乏科学的计划和管理。可以说，以这样的方式开发软件，不仅效率低下，还会导致软件产品质量的低劣。

《小组软件开发过程》（简称“TSPi”），是美国 Embry-Riddle Aeronautical 大学为计算机科学系研究生和高年级本科生开设的一门软件工程课的教科书。本书作者 Watts S.Humphrey 教授是软件工程方面的权威，有多年的研究、实践和教学经验。这本书系统地论述了如何以开发小组的形式来进行软件的开发，并对开发过程作出了具体而详尽的指导，包括小组成员之间的协调、进度的管理、质量的控制等，也许是令读者最感兴趣的各个方面。我们把这本书介绍给国内读者，希望能够对广大的软件开发人员尤其是项目经理起到一定的参考作用，也算是为我国软件产业的发展尽一点微薄之力。

由于时间较紧，加之译者水平有限，书中疏漏和不当之处在所难免，敬请读者和专家批评指正。

如今，软件开发越来越复杂，工业上对更好的软件工程师的要求也越来越迫切。这就意味着，我们在软件开发教学中必须使学生掌握有关软件工程实践的知识和技能。Embry-Riddle Aeronautical 大学工业顾问部已经为初级软件工程师制订了下面的几条标准：

- 与他人和外界的交流（包括口头和书面上的）；
- 作为开发小组中的一员进行工作的能力；
- 能够胜任软件开发的前期工作（需求和高级设计）；
- 对工作的敬业精神；
- 应用软件过程中的知识和技能；
- 具有一定的计算基础；
- 具有宽广的知识面（学习新技术的能力）。

在我教授软件导论课程的八年经历中，我曾经试图向学生们展示软件开发的前景，并且要他们作为开发小组的一部分进行工作。我已经做过了许多实验，包括真正的工程和玩具工程、小规模小组开发和大规模小组开发、广泛使用工具和几乎不使用工具、强调产品和强调过程等。但是大部分时候，我都试图要做太多的事，结果却是教师和学生都以灰心和失望告终。而直到最近，我才有了一点有限的成就。

去年，我使用了《小组软件开发过程》（TSPi），结果出乎我的意料，我取得了自从实验以来最大的成功。虽然 TSPi 并不是一件万灵法宝，但它确实惊人地增加了我们软件工程教学的效果。不仅是对学生，TSPi 还告诉老师该做什么，该如何做，以及该什么时候做。这本教材涵盖了全部必须的 TSPi 资料：文字、表格和有关学生小组软件开发的几乎所有方面的指导。它出色地诠释了 TSPi 的内容，而且还为每一个小组成员应担任的角色进行了完整的描述。关于如何解决小组内部管理的问题，它给出了常规的意见；在计划、跟踪和评估小组成员的成果和品质方面，它列出了量化的技术标准。

虽然本书只包括了关于两个工程需要的初步阐述，但 TSPi 具有足够的灵活性，完全可以对各种适当规模的工程应付自如。TSPi 可以应用于维护性的工程，应用于需求/设计主题，还适用于任何小组软件开发行为。例如，我们现在在高级设计课上就是使用 TSPi 来为真正顾客开发产品的。这样，TSPi 把一项可靠的软件开发计划和出色的教学计划合并在一起了。开始的时候，学生们在学习 TSPi 的过程中，习惯于与小组一起工作。一两个阶段以后，这个小组就可以根据他们先前的经验来提高自己的水平了。

在使用 TSPi 进行软件开发工程这一学科中有许多东西要学，但无论是通过从前的学习还是自学，学生们都必须具有个人软件设计（PSP）的初步经验。在 Embry-Riddle，我们在一年级程序设计课上向学生介绍 PSP，这就为学习和应用 TSPi 做好了准备。不管是对于教师还是学生，TSPi 课程都需要大量的时间。在几次民意调查中（每一个阶段的末尾），学生们都非常认同 TSPi。最普遍的抱怨是关于资料的收集和记录工作中的数据——虽然他们绝大多数人也承认他们明白这一项的重要性。

一句话，如果你曾经努力要上一门高质量的软件工程计划课的话，我竭力向你推荐 TSPi。它提供了对学生们传授软件工程实践经验的指导、说明和支持，使他们更好地为未来的工作作准备。

Thomas B.Hilburn
Embry-Riddle Aeronautical University

为了帮助我准备这个给学生们的前言，Tom Hilburn 教授从他在 Embry-Riddle 的第一届 TSPi 课的学生中选出了 3 个人。他们中的两个曾在工业部门搞过封闭开发，而 3 个人全都是他们开发小组里的领导者。当被问到关于这门课本身时，Tom 要他们说出他们想对同学们说的话；下面我就引用几段。

为什么要学 TSPi？为什么要在这个连工作本身的耗时都比填一大堆表格来得少的时代应用 TSPi？这些问题当然都有一个非常好的答案。当你学习新的知识时，你不会想要找一种你自己不可能学会的东西来学习。你一定是先学一些比较容易的知识以便了解当前这一学科的发展情况。而这，就是你学习 TSPi 的方法——你得从一个不需要加工处理的小工程开始。一旦你掌握了 TSPi，你就会意识到这是必要的。虽然 TSPi 在大多数学校中不需要学习，但它在庞大的工业工程中是必须的。这就是 TSPi 的用武之地——它可以帮助每个人更了解自己的职责，使每个人都知道该如何完成这项工程。所以要记住，现在你刚刚学习了这个开发的过程，而这种知识的益处在你遇到个人无法完成的工程之前是无法显现出来的。

绝大多数计算机科学系的学生都是自学如何进行程序设计开发的。在他们的小组工程中，他们总喜欢用结构化的步骤来帮助他们应付下一个小组的开发任务。小组内的交流成了软件开发过程中一个全新的方面。像相互之间交流、信任、动机、解决问题、承担义务、奉献精神、工作量的平衡、个人角色确定、同事之间的感情、权力问题、向同组人学习工作方法等等，都是影响小组工作进展的新因素。TSPi 为解决这些通常的问题提供了行之有效的意见。学生们必须适应他们的每个小组的不同条件，学习如何解决课本上没有提到的其他问题。

目前，小组开发工作的关键是，小组成员之间必须要有良好的交流和信任，以及必须具备与自己不熟识的人一起工作的能力。绝大多数小组都是希望尽快完成工作的，这样就可以有大量的时间来进行对软件质量的测试和检查。小组成员们不但要在他们的工作适合自己时做好承担更多任务的准备，而且还要准备分担工作负担过重的成员的工作。这种工作模式鼓励我们决不能让小组工作陷入困境，无论何时何地，只要我们能够提供帮助，我们就应当伸出援助的手，特别是在工程的最初阶段，我们的小组更倾向于尽快做出产品，而不是遵循应当遵循的步骤的时候。接下来的阶段，工程师们就会看到计划和组织的必要性，从而就会更倾向于遵循应有的步骤。

我们一般很少会有什么关于工作动机和承担义务方面的问题，但是这种问题一旦出现，我们就不得不去扼要地讨论这种问题。最困难的事情莫过于在有关小组其他成员的决定和处理问题中保持客观了。如果我们没能做到经常性的小组成员交流，这个工程就很有可能事倍功半。我们应当了解小组工作的责任、小组成员之间的关系，以及互相帮助的重要性。同时，我们也必须懂得，只要我们真正为此付出了努力，整个小组都会乐于帮助我们。TSPi 给了学生们一个良好的开端，使学生们真正了解将来他们在工业中需要做的一切。将来，学生培养计划会更好地进行组织，以便加入这样的课程。

在工程管理方面要注意：TSPi 需要卓越的管理技巧和出色的能力。从 TSPi 中可以学到许多可应用于小组工作的东西，但是只有小组的领导者才能负责工程的总体规划。作为小组领导者，我曾经尝试过许多管理措施，大部分都很有效，但是至少有一次没有任何效果。作为领导者，你必须乐于投入和你的同伴们同样多的时间，深入底层工作，而且要在需要帮助的时候伸出援助的手。一个领导者必须是能够激励他人的人，语言和行为都要做到这一点。给你的小组成员们工作的空间，在设计中要给他们有创见的机会，在出现问题和挫折时领导大家一起渡过难关。一定要尽可能地让整个小组参与计划的制订，让每一个成员参与每一个步骤，同时要监视小组计划时间表，当出现问题时及时采取措施。

小组领导者必须做到防患于未然。要知道，小组成员可能不肯全心全意工作，也有可能因为突发疾病或受伤而耽误了进度。这些情况在我负责的工程中都是发生过的。这种问题既有可能使整个小组分裂、降低小组的工作效率，但也有可能使整个小组更加团结一致，提高小组的生产率，这就全靠你如何应付这样的局面了。这时，也就更需要你作为领导的愿望和技巧了。珍惜小组工作的经历，努力去了解你的小组，尽力让大家充满信心；工作要努力而有计划；学会用电子图表来从零碎时间中节省时间；保持小组成员之间的顺利交流，大家经常会面。记住：不管情况多么糟糕，都会有办法的，“永不退缩，就永不失败。”

Celeste Berry, Ryan Hoppes, Marc Lovelace
Embry-Riddle Aeronautical University

这本书是为已经学过或是使用过个人软件程序设计（PSP）的学生或工程师们准备的。你可能在本科、高级专科或是早期导论课程中学习过 PSP。或者，你也可以是一个曾经在实际工作中研究过 PSP 的富有经验的工程师。无论如何，只要你掌握了 PSP，你就具备了使用本书中方法和实践的背景。

在你学完 PSP 后，你很可能需要有关如何在软件开发过程的许多任务上应用它的指导。这就是小组软件开发过程（Team Software Process，即 TSP）的主旨：在软件开发过程中提供一个用来可靠地应用工程学方法的框架。

关于小组工作有很多内容可讲，这本书涵盖了最基本的内容。TSPi（the introductory Team Software Process）介绍了小组的概念，按部就班地指导你去创建一个小组，并在小组里工作。但是要注意，这只是一门导论课程，它不可能包含你将来进行大规模产业计划时需要用到的全部资料。

一、TSPi 是如何帮助工程师的

这本书将向工程师们介绍协作软件开发的知识。TSPi 提供了一整套结构化的步骤，告诉工程师们每一步该怎样做，并且详细阐述了把这些步骤连接起来以完成一个完整的产品的方法。TSPi 也提供了两个很有趣而且具有适当的挑战性工程计划练习，每一个都是小到只有几周就能完成、大到可以模仿一个典型的小工程计划。如果有能力的工程师们按照本书中的指导去做，他们一定会做出一个工作成品来的。

在 TSPi 的策略中，小组在 2 到 3 个周期内就可以研制出产品来。在第一个周期中，小组创建出一个小的工作产品内核。在接下来的周期内，再向这个内核上加入新的功能。这个策略体现了使用先前工程数据来规划新的工程的好处。同时，在每一个周期中工程师们都有新的任务，这样在一个工程中每人都可能有两到三次不同的工作经历。几个开发周期过后，工程师们就会对联合协作开发有一个充分的了解，他们也会更加愿意在自己的工作中使用 TSPi。

二、为什么我们要学 TSPi 这门课程

在培养从事软件工程专业的学生的过程中，工程学课程已经被证明是很有效果的。因此，许多大学现在都开这类课程。而想学这些课程的学生经常都是比开课时要多，学生们收集他们将来工作中可能用到的资料，他们发现，有关协作开发的课程恰好能够满足他们的需要。等他们毕业后，学生和雇主都认为，软件工程课是为将来在产业部门工作的一个良好基础。

现在，对于协作开发工程课程我们已经有不少经验。虽然许多这类的课程是成功的，但普遍有三个问题：第一，学生经常去尝试过于庞大的工程；第二，他们经常只注重结果而忽略过程；第三，总有一个或更多的小组成员闹分裂。虽然 TSPi 不能完全避免这些问题，但它提供了关于如何避免或消除这些问题的指导。

为了更充分地利用课上时间，小组软件开发课程应当基于已经经过了实践验证的工程，而且必须经过仔细的策划。如果没有详细的操作步骤和现成的小组框架结构，工程师们就只能独立地设计整个过程，使他们的工程计划付诸实施。而且在上述条件下，这些工作小组也不得不在不断地尝试和碰壁过程中去摸索小组创建和协同工作的基本原则。这显然是代价高昂而且很不必要的，因为协同工作的原则本来简单易懂、众所周知。

TSPi 教给工程师们的是行之有效的协同工作方法。它从小组创建开始，然后就在开发产品的过程中应用这个标准的框架。只要工程师们有 PSP 的基础，他们就能按照 TSPi 的基本原则来对自己的工作进行计划和管理。应用 TSPi，可以使工程计划变得更有效率，使工程师们得以在软件开发本身集中精力，而不是在小组创建和管理这类事情上浪费大量时间。

TSPi 为小组中每一个成员都提供了确定的角色。每一个角色都有明确的任务，以及每个分计划需要完成的时间。当小组里的每一个成员都知道自己和其他人该做些什么时，他们就可以作为一个有效率的团体来进行工作了。如果某个成员没有完成任务，其他的成员可以了解到这一点，再及时地处理所遇到的问题。当小组自己无法处理个人的问题时，他们就应当向他们的教师或管理人员求助。这本书中的“给教师的话”就提供了许多解决这类常见问题的方法。

当学生小组成员分工明确、责任清楚后，教师就可以公平明确地打分了。每个学生都

有个人表现和整体结果两方面的分数等级。这不但可以鼓励学生们创新，同时也是一个给出学科成绩的公平办法。

三、本书的组织结构

这本书是为了使学生掌握 TSPi 的过程和步骤而设计的。开头两章（第一部分）是对理论的简单介绍。第二部分中的章节则是整个小组研究周期的详细内容。本书详细解释了小组软件开发的步骤，并且给出了 TSPi 完整形式的例子。

第三部分提供了小组成员角色的细致描述：小组领导者，开发经理，计划经理，质量/生产经理，以及技术支持经理。读过这一部分中有关你个人角色的内容后，你就可以在工作中把书中所介绍的内容作为参考了。

TSPi 课程开始的时候，每一个学生都要完成一份介绍个人兴趣和背景的 INFO 表格。教师将根据这些信息把整个班级分成 5 个工程小组，再指定每个小组成员的初步角色。如果一或两个小组有 4 或 6 个组员，教师就必须作一定的角色调整。所有的角色都要有人充当，而每个工程师必须至少扮演一个角色。对于一个有 4 名工程师成员的小组，其技术支持经理必须是从组员中选出的；对于有 6 名工程师成员的小组，质量/生产经理则必须分成两个：质量监督经理和生产监督经理。

小组选好、角色确定之后，各个小组就开始他们的工程计划，并且汇报进度。在每一个开发周期结束时，组内工程师们都要对整个小组的总体表现和各个角色的表现进行评估。根据这些信息，教师就可以对他们的工作作出评价，从而在下一个开发周期中更好地分配小组角色。如果有必要的话，教师还可以进行组间成员调动，但如果没什么严重问题的话，每个小组在整个课程期间是不会有变动的。

四、使用预先设计的标准课题

虽然 TSPi 可以应用于几乎任何一个工程，这本书只是提供了两个预先设计的标准课题，它们是为满足将来可能遇到的各种情况而设定的。虽然在这里使用面向顾客的实际课题可能有一些好处，但是出于 3 个原因，我们不推荐这样做。第一，课程本身有严格的时间表，即使大部分顾客起初都可能同意按照固定时间表来进行，也不大可能有几个顾客真正知道开发软件究竟需要多少时间。而且，通常一开始工程师们还不知道该如何依照严格的时间表来管理规划整个工程，工程失败的可能性是很大的。与这个问题同时出现的麻烦是，真正的顾客需求都是不稳定而且很不明确的，这又会导致经常性的变化，耽误大量时间。

使用预先设计的标准课题的第二个原因是，一门协同工作课程是为教授专门课程来设计的。虽然工程的目标之一是要开发出一种产品，但基本的课程目标却是要阐述使用这些经过实践检验的软件工程方法的好处。对于一项真正的面向顾客的工程计划，优先考虑的就是要满足顾客的需求。当顾客需求有所改变或者顾客占用时间来咨询的时候，整个工程就得暂时停顿下来。这样，完成任务的时间表就不得不压缩，小组经常会把精力集中在完成产品上，忽视了开发的过程。结果，我们的课程就只能使学生们学到怎样不科学地开发软件了。

第三个使用预先设计的标准课题的理由是便于小组之间的比较。有了同一个课题的不同实施办法，所有的小组都可以参与等级评估。每一个小组都可以介绍他们的开发步骤，回答关于他们设计实现的问题，介绍他们计划的执行情况，以及检验的情况。这种方法可以充

计开发的工作周期则需要 4 到 5 周。这样，一个只包括最起码内容的完整工程最短周期需要 6 到 7 周。表 P.2 列出了一个开发一系列用户需求功能工程的协同工作计划。类似地，设计开发工程由表 P.3 来表示。本书也可用于半个学期的教学中，这样 3 个开发周期的课程就要 15 周，两个开发周期的课程就要 11 周，而一个开发周期的计划只要 7 周。

不管如何使用本书，标准 TSPI 都将指导学生们组织自己的小组，制订计划并执行计划。如果一个小组没有任何关于 TSPI 的经验，那么他们很可能无法在少于 3 到 4 个星期的时间内完成任何一个开发周期，那是因为新组员学习适应小组开发步骤和学会如何协同工作需要一定的时间。这也就是第一个 TSPI 开发周期需要 7 个星期的原因。事实上，在接下来的周期内，我们就只需要 4 个星期了。

表 P.2

一个 TSPI 短需求工程

周次	周期 1
1	启动, 计划
2	需求
3	需求
4	最终检查

表 P.3

一个 TSPI 短设计工程

周次	周期 1
1	启动, 计划
2	需求
3	设计
4	设计
5	最终检查

六、课程准备

本课程的先修课是 **PSP**（个人软件程序设计），导论课程或是本科课程都可以。如果学生是在几个学期之前学习的 **PSP**，那么他们就应当已经在中间课程中使用过 **PSP** 了。如果他们没用过，他们就得通过一到两次复习课来重温一下 **PSP** 计划、数据采集和质量管理方面的知识。当然，没有 **PSP** 经验或是经验很少的学生在整个课程中都需要仔细的指导。

在开始一项小组计划前，学生们必须有软件设计和软件需求的知识背景。配置管理、计划管理和软件测试也很有用。学生们必须能够熟练使用编程语言和工具。

Watts S.Humphrey
Sarasota,Florida

第一部分 绪论

第一章 TSPi 简介	3
1.1 什么是 TSPi	3
1.2 TSPi 原则	4
1.3 TSPi 的设计	5
1.3.1 提供一个在 PSP 基础上建立的简单框架	5
1.3.2 把产品的开发划分为数个周期	5
1.3.3 建立标准的质量和效率评估机制	6
1.3.4 为小组和学生们提供明确的评估标准	6
1.3.5 进行角色和小组评估	6
1.3.6 必要的开发纪律	6
1.3.7 提供协同工作指导	7
1.4 TSPi 的结构和流程	7
1.5 TSPi 过程	10
1.6 课本结构和流程	10
1.7 小结	11
第二章 小组软件开发逻辑	13
2.1 为什么工程会失败	13
2.2 常见小组问题	14
2.2.1 领导不力	14
2.2.2 无法协调或合作	15
2.2.3 缺少参与	15
2.2.4 拖延和缺少自信	15
2.2.5 质量低劣	15
2.2.6 随意增加功能	15
2.2.7 无效的互评	16
2.3 小组是什么	16
2.3.1 小组规模	16
2.3.2 具有凝聚力的小组	16

2.3.3 协同工作基本条件	17
2.4 建立高效的小组	17
2.4.1 小组内聚力	17
2.4.2 挑战性的目标	17
2.4.3 反馈	18
2.4.4 共同工作框架	18
2.5 小组是怎样发展的	18
2.6 TSPi 是如何创建小组的	19
2.6.1 目标	19
2.6.2 角色	19
2.6.3 计划	20
2.6.4 交流	20
2.6.5 对外交流	20
2.7 小结	21

第二部分 TSPi 过程

第三章 启动一个小组工程	25
3.1 为什么需要小组启动过程	25
3.2 小组目标	25
3.2.1 制定目标需要考虑的事项	26
3.2.2 制定小组目标	26
3.2.3 TSPi 的目标制定	26
3.3 小组成员目标	28
3.4 角色目标	29
3.4.1 小组长目标	29
3.4.2 开发经理目标	29
3.4.3 计划经理目标	30
3.4.4 质量/生产经理目标	30
3.4.5 技术支持经理目标	30
3.5 TSPi 启动流程	30
3.5.1 学生信息	31
3.5.2 产品目标	31
3.5.3 小组分配	31
3.5.4 小组目标	32
3.5.5 小组会议和第一次小组会议	36
3.5.6 数据需求	36
3.5.7 工程开始	36
3.5.8 项目手册	36