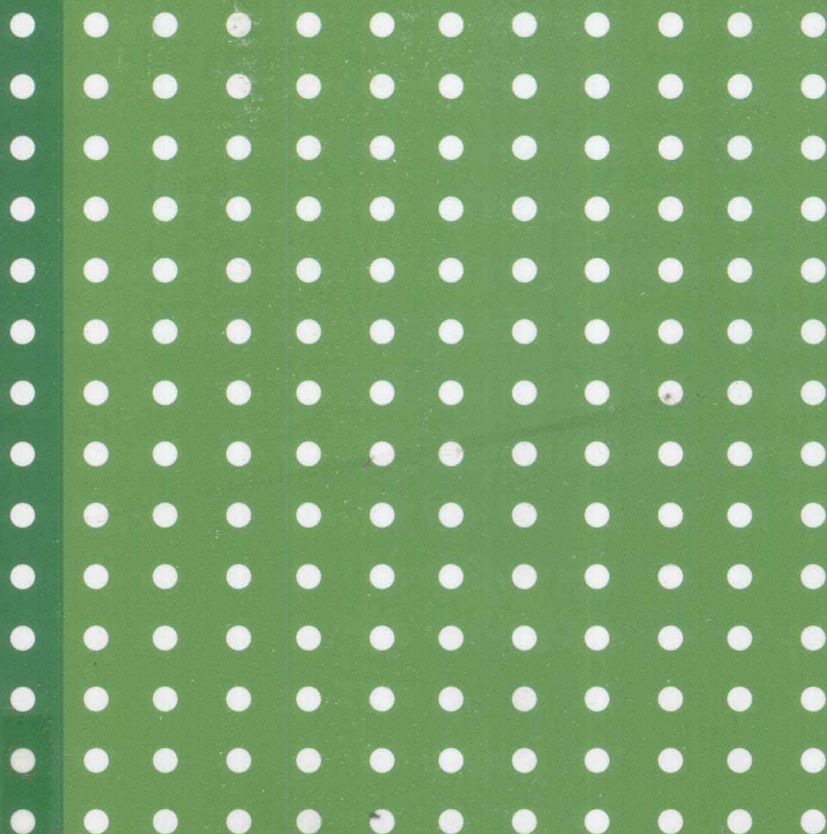


重点大学软件工程规划系列教材

软件工程项目实训教程

——基于微软VSTS

张家浩 主编



清华大学出版社

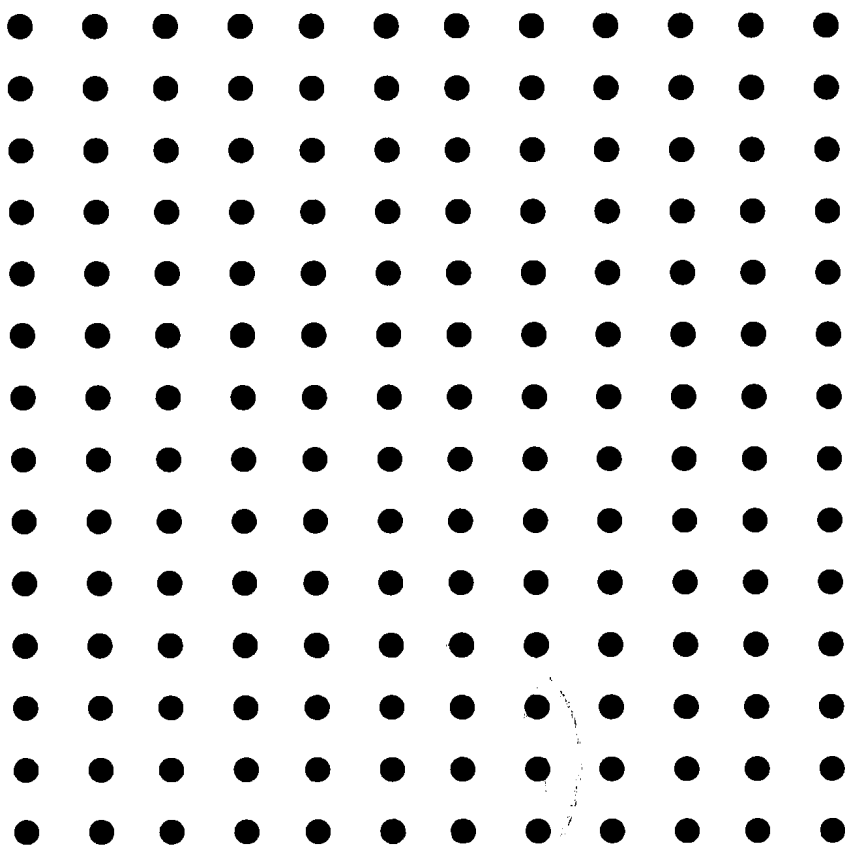


重点大学软件工程规划系列教材

软件工程项目实训教程

——基于微软VSTS

张家浩 主编



清华大学出版社
北京

内 容 简 介

本书在简单回顾软件工程理论和技术方法的同时,介绍基于 VSTS 的软件过程管理方法,并带领学生实际使用这些管理工具。全书采用“ATM 扩展”项目作为案例。每章还配有课堂作业与问题思考,方便老师和学生使用。

本书可作为软件学院或软件工程专业的软件工程实践教材,也可作为其他相关专业的实践教学用书,还可作为从事软件开发的科技人员的参考书、培训教材等。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

软件工程项目实训教程——基于微软 VSTS/张家浩主编.--北京:清华大学出版社,2011.6
(重点大学软件工程规划系列教材)

ISBN 978-7-302-24895-8

I. ①软… II. ①张… III. ①软件工程—高等学校—教材 IV. ①TP311.5

中国版本图书馆 CIP 数据核字(2011)第 033300 号

责任编辑:付弘宇 赵晓宁

责任校对:白 蕾

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954,jsjjc@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015,zhiliang@tup.tsinghua.edu.cn

印 装 者:北京市清华园胶印厂

经 销:全国新华书店

开 本:185×260 印 张:16.25 字 数:394 千字

版 次:2011 年 6 月第 1 版 印 次:2011 年 6 月第 1 次印刷

印 数:1~3000

定 价:27.00 元

产品编号:040382-01

前言

本书是什么,不是什么

本书是给软件学院软件工程专业高年级或研究生的“软件工程”课程配套的实践书,它是一个动手的“脚本”,是更适合在“机房”里、对照着屏幕“操作”的书。基本上不是仅仅在课堂上讲,在图书馆里看的书。

如果在“软件工程”课程之外,专门开有“软件工程实训”课程(软件学院暑期通常为高年级或研究生专门开设此类课程),则可以直接使用本教程。如果并非实训课程,则作者给任课教师以下建议:以理论为主、实践为辅的,可选作者已出版的《现代软件工程》(机械工业出版社,2008年8月)为主教材,本书为实践辅导教材。

既然本书只是一个“脚本”,因此,在不同的导演(老师)和不同的演员(学生)手中,当然会产生出完全不同的效果。本书作者使用此讲稿,已经在本科高年级和研究生中讲过4次,每次也都有新的变化。因此,本书只是提出一些思路,任课教师不必受本书的约束,这也是作者所希望的。

本书如何取舍

IEEE的《软件工程知识体系指南(SWEBOK 2004)》有11个知识领域、PMI的《项目管理知识体系指南 2008 第4版(PMBOK 2008)》定义了9个知识领域、42个过程,CMM/CMMI有5级,软件质量过程有6个关键活动……软件工程涉及内容包罗万象,必须进行适当的裁剪。取舍的原则只能是根据实训条件(包括时间)和学生可接受程度。

本书有何新意

有关“软件工程”的实践教程,图书云集,著者众多。作者跻身其中,若无一点新意,很怕滥竽充数,自取其辱。

“案例教学”方式已少新意;采用VSTS平台也算不上太多创新;若能将学生的项目实践活动真正按照一个规范(并非书本上的规范,而是业界流行,下同)的软件工程过程模式进行,并在规范的软件过程管理工具和平台的支撑与控制下完成,使学生在完成项目开发的同时,甚至项目开发目标都在其次,更重要的是能够真实、具体地体验一遍现代软件工程的开发管理过程,这才算有点新意,这样,才能算得上是“软件工程”的项目实训(重点是“实”)。

作者的另一个目的,就是把“软件工程”课程的教学看成是一个项目(某个学期、某位老师上某门课,应该可以算是一个项目)。希望通过尝试,通过成功或失败的实际案例,从课程目标设置、过程设计、内容安排,直至成果检验,提出一个教学项目的“目标与过程管理”的可行方案。希望(至少在这门课上)从此不要再搞那些无作用的“督导”、无聊的“网评”了吧!

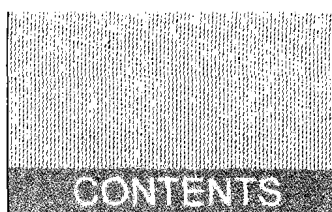
如果做不到这些,可能我们都无颜站在讲台上教学生什么软件工程了,用“作坊”的方式能教出“现代软件工程”的能手?我不相信。

是为前言。

张家浩

2011年2月

于南京百合果园



目 录

第 1 章 软件工程实训项目课程导论	1
1.1 软件工程与我	1
1.1.1 角色认知：我的未来不是梦	1
1.1.2 课程定位：软件车间主任的培训班	3
1.1.3 实训与实战：态度决定一切	4
1.1.4 理论与实践：相映成趣、相得益彰	5
1.2 实训各阶段的任务与要求	6
1.2.1 阶段 1：组成项目团队	6
1.2.2 阶段 2：提出项目目标	6
1.2.3 阶段 3：制定项目计划	7
1.2.4 阶段 4：完成需求定义	7
1.2.5 阶段 5：设计系统架构	8
1.2.6 阶段 6：完成代码开发	8
1.2.7 阶段 7：提交测试验收	8
1.2.8 阶段 8：进行项目总结	9
1.3 实训项目课题的选择	9
1.3.1 关于实训项目课题的选择	9
1.3.2 实训项目案例——ATM 系统扩展	9
1.3.3 在相同基础上的个性化发挥	10
1.4 带着兄弟们上路	10
1.4.1 组成项目团队	10
1.4.2 从角色需要考虑人选	10
1.4.3 准备 VSTS 环境	11
1.4.4 阅读教材与参考书	12
1.5 本阶段小结——组成项目团队	13
1.5.1 理解实训项目的课程目标	13
1.5.2 了解如何进行课程阶段成果的检查	14
1.5.3 本章的理论基础和实践内容小结	14
1.6 本章作业与问题	14
1.6.1 本章作业	14
1.6.2 问题：进一步的思考	14

第2章 实训项目的选题与目标范围确定	15
2.1 认识我们的目标	15
2.1.1 微软创新杯	15
2.1.2 大赛主题与选题	16
2.1.3 理解大赛主题	16
2.2 有一个好的点子	17
2.2.1 基于传统问题域的创新选题	18
2.2.2 关注新的应用领域	19
2.2.3 寻找新的技术应用热点	20
2.2.4 参考各类软件创新大赛的获奖选题	20
2.2.5 从既往的项目中选择课题	21
2.3 选题的目标设计与价值评价	21
2.3.1 目标与方案	21
2.3.2 最初的目标：大赛通知	22
2.3.3 第一个交付成果：《项目计划书》	23
2.3.4 再谈目标——大赛的评价标准	25
2.4 课题的难度与可行性	27
2.4.1 何谓“难度”	27
2.4.2 难度与可行性的关系	27
2.4.3 可行性的非技术性考虑	28
2.5 确定项目的范围	29
2.5.1 为目标而确定范围	29
2.5.2 考虑条件和资源	29
2.5.3 修剪你的“范围树”	30
2.6 项目团队组建与项目初步规划	30
2.6.1 角色与分工	31
2.6.2 项目总体规划	31
2.6.3 为第一次提交，制订更详细的工作计划	32
2.7 实训项目案例——《ATM 扩展项目计划书》	32
2.7.1 参赛作品构思的创意与价值	32
2.7.2 参赛作品的目标实现形式	33
2.7.3 参赛作品目标实现的可行性	34
2.7.4 团队组成与角色分工	34
2.7.5 项目时间进度表	34
2.8 本阶段小结——通过项目初审	35
2.8.1 软件创新大赛各阶段评审重点	35
2.8.2 第一轮入围与评判标准	35
2.8.3 本章的理论基础和实践内容小结	36

2.9	本章作业与问题	37
2.9.1	本章作业	37
2.9.2	问题：更进一步的思考	37
第3章	交付过程模型与项目管理控制	38
3.1	交付过程模型与过程管理	38
3.1.1	过程模型的一般意义	39
3.1.2	微软公司的软件过程模型 MSF 与 VSTS	41
3.1.3	统一过程模型 RUP 与 IBM 的 Jazz	44
3.1.4	MSF 与 RUP 的比较	47
3.2	交付过程模型的结构与关键行为	47
3.2.1	工作项与工作产品	48
3.2.2	角色	48
3.2.3	流程	48
3.2.4	活动和步骤	49
3.3	为实训项目搭建 VSTS 平台	49
3.3.1	VSTS 的逻辑结构与物理结构	49
3.3.2	安装 VSTS 系统的服务器端	50
3.3.3	安装 VSTS 系统的客户端	55
3.4	使用 VSTS 定义项目	56
3.4.1	在 VSTS 上定义项目	56
3.4.2	使用团队资源管理器定义项目的工作项内容	57
3.5	为项目定义基线与状态	62
3.5.1	基线与状态控制	62
3.5.2	用 VSTS 设置项目的基线	63
3.6	创建门户与团队报告	65
3.6.1	打开团队项目门户	65
3.6.2	自定义并扩展团队报告	66
3.7	实训项目案例——ATM 扩展	66
3.7.1	定义前的准备	66
3.7.2	为 ATM 扩展项目选择生命周期模型	66
3.7.3	为 ATM 扩展项目定义项目计划	67
3.7.4	为 ATM 扩展项目定义具体的工作项内容	67
3.7.5	为 ATM 扩展项目定义基线	68
3.7.6	查看 ATM 扩展项目的团队门户和团队报告	68
3.8	本阶段小结——召开第一次项目例会	68
3.8.1	如何开好一次项目例会	69
3.8.2	本章的理论基础和实践内容小结	69
3.9	本章作业与问题	69

3.9.1	本章作业	69
3.9.2	问题：更进一步的思考	69
第4章	需求工程中的需求开发与管理	71
4.1	软件需求的获取与描述	72
4.1.1	需求获取阶段的工作目标与关键交付物成果	72
4.1.2	基于 UML 的电梯控制系统需求模型	73
4.1.3	项目案例：ATM 扩展项目的需求获取过程	75
4.1.4	项目案例：ATM 基本系统的业务用例模型	76
4.1.5	项目案例：在 ATM 网络系统中加入“前置机”的功能与作用	77
4.1.6	项目案例：ATM 系统的扩展以及相关的用户确认	81
4.2	需求分析模型与关键需求	82
4.2.1	需求分析阶段的工作目标与关键交付物成果	82
4.2.2	项目案例：ATM 基本系统的需求分析模型	83
4.2.3	项目案例：ATM 扩展的需求场景	84
4.2.4	项目案例：ATM 扩展的关键需求分析	86
4.2.5	项目案例：ATM 关键需求实现的用户验收	86
4.3	文档化的需求处理与需求规格描述	87
4.3.1	需求处理阶段的工作目标与关键交付物成果	87
4.3.2	《需求规格说明书》的主要内容	88
4.3.3	项目案例：ATM 项目的《需求规格说明书》	88
4.4	Borland Caliber 的需求定义与管理功能	89
4.4.1	Borland Caliber 的需求定义与管理概念	89
4.4.2	Borland Caliber 的需求定义与管理过程	90
4.5	使用 Caliber 定义项目需求	93
4.5.1	创建新项目	93
4.5.2	为项目创建需求树	93
4.5.3	定义需求信息	94
4.5.4	为需求分配属性值	95
4.5.5	为需求分配用户和组	96
4.5.6	链接参考文档	96
4.5.7	创建追踪能力链接	96
4.5.8	定义需求验证过程	98
4.6	使用 Caliber 对需求进行基线管理	98
4.6.1	查看项目的需求基线	98
4.6.2	创建项目的需求基线	99
4.6.3	初始化项目的需求基线	101
4.6.4	锁定项目的需求基线	101
4.6.5	项目基线的电子签名	102

4.6.6	添加项目基线的电子签名	103
4.6.7	查看项目基线的签名报告	103
4.6.8	比较项目的需求基线	103
4.6.9	有关项目需求基线的小结	104
4.7	将 Caliber 与 VSTS 集成	105
4.7.1	安装 Caliber 的 VSTS 插件	106
4.7.2	让 Caliber 与 VSTS 工作项关联	107
4.7.3	在 VSTS 中创建 Caliber 的需求项	108
4.7.4	建立 Caliber 的需求项与 VSTS 工作项之间的跟踪关系	109
4.7.5	总结: Caliber 与 VSTS 一起工作	109
4.8	在 Caliber 上建立 ATM 扩展需求	109
4.8.1	从业务用例模型到 Caliber 需求树	109
4.8.2	了解 Caliber 上的 ATM 需求	110
4.8.3	在 Caliber 中扩展自己的 ATM 需求	111
4.8.4	在 Caliber 上确定项目的需求范围边界	111
4.8.5	与 VSTS 工作项相关联并确定基线	111
4.9	本阶段小结——通过需求评审	112
4.9.1	理解实训项目的需求评审要求	112
4.9.2	开展实训项目的需求评审活动	113
4.9.3	本章的理论基础和实践内容小结	113
4.10	本章作业与问题	113
4.10.1	本章作业	113
4.10.2	问题: 更进一步的思考	114
第 5 章	基于关键质量属性的架构设计	115
5.1	最初的架构模型设想	115
5.1.1	从需求模型开始	116
5.1.2	一般架构模型的基本考虑	117
5.1.3	从需求模型到架构模型的转换	118
5.1.4	电梯控制系统的架构设计	119
5.2	关注与架构有关的关键质量属性	121
5.2.1	满足关键质量属性需求的架构设计	121
5.2.2	电梯控制系统关键质量属性需求分析	122
5.2.3	规范的关键质量属性场景描述	126
5.2.4	有关 ATM 扩展的关键质量属性场景描述	126
5.3	基于关键质量属性需求的分析与架构设计	127
5.3.1	“可用性”需求的现状分析	127
5.3.2	实现“实时性”需求的对策	133
5.3.3	实现“实时性”需求的方法	134

5.3.4	基于关键质量属性的架构分析	135
5.3.5	实时故障恢复系统的架构设计考虑	136
5.3.6	实时故障恢复系统的详细设计考虑	137
5.4	搭建一个基于 MVC 模式的 Struts 架构	138
5.4.1	选择 MVC 模式的理由	139
5.4.2	用 Struts 搭建一个“轻量级”的应用架构	141
5.4.3	更进一步地体验 Struts 架构中的 MVC 组件	142
5.4.4	项目作业：比较在 Struts 架构上搭建 ATM 系统的优劣	144
5.5	在 Struts 架构上运用面向对象设计模式	145
5.5.1	业务处理流程灵活性的质量属性场景描述	145
5.5.2	实现业务处理流程灵活性的战术对策	146
5.5.3	采用工厂方法实现流程灵活性的关键质量需求	147
5.5.4	在 Struts 框架下实现设计模式的应用	148
5.6	使用 VSTS 可视化的分布式系统设计器构建系统架构	150
5.6.1	分布式系统设计器的作用和相互关系	150
5.6.2	定义组件的提供者	152
5.6.3	定义对组件提供者终结点的控制	153
5.6.4	定义组件之间的连接	154
5.6.5	应用程序的实现	155
5.6.6	项目作业：使用 VSTS 应用程序设计器实现 ATM 系统	157
5.7	架构文档与架构评审	157
5.7.1	规范的架构设计活动过程与制品	157
5.7.2	需要编写的架构视图和文档	158
5.7.3	透过架构视图表现架构设计的核心内容	160
5.7.4	针对一般要素的架构设计评审	161
5.7.5	针对关键质量属性需求的架构设计评审	161
5.7.6	ATM 实时故障恢复系统的构架设计评审	162
5.8	本阶段小结——通过架构设计评审	165
5.8.1	理解项目实训课程的架构设计评审要求	165
5.8.2	开展实训项目的架构评审活动	166
5.8.3	本章的理论基础和实践内容小结	167
5.9	本章作业与问题	167
5.9.1	本章作业	167
5.9.2	问题：更进一步的思考	167
第 6 章	代码开发阶段的软件过程控制与管理	168
6.1	用 VSTS 实现对源代码的控制与管理	168
6.1.1	从建立规范的源代码开发管理环境开始	168
6.1.2	使用源代码管理器对个人的工作区进行管理	169

6.1.3	向存储库和工作区中添加文件夹/文件/解决方案	171
6.1.4	通过配置签入/签出策略设置对变更活动的约束	173
6.1.5	签出	174
6.1.6	签入	175
6.2	使用 TFVC 进行版本控制	175
6.2.1	设置团队版本控制环境	176
6.2.2	决定控制什么和由谁来进行控制	176
6.2.3	在开发中使用 TFVC 进行版本控制和管理	177
6.3	使用 TFB 进行构建与发布管理	177
6.3.1	什么是现代构建与发布管理	177
6.3.2	安装并配置 TFB	178
6.3.3	使用 TFB 进行生成与每日集成	181
6.3.4	查看 TFB 报告	181
6.4	体验 VSTS 的单元测试与测试管理	182
6.4.1	测试的概念与 VSTS 的测试功能	182
6.4.2	单元测试的概念	184
6.4.3	建立本地的单元测试环境	185
6.4.4	为单元测试设置 VS 项目	187
6.4.5	使用测试管理器运行和管理单元测试	189
6.4.6	尝试 VSTS 的测试代码覆盖	189
6.4.7	运用 VSTS 托管代码分析工具	191
6.4.8	将测试与 TFS 集成并发布测试结果	192
6.5	本阶段小结——通过代码评审	194
6.5.1	理解实训项目的代码评审要求	194
6.5.2	开展实训项目的代码评审活动	194
6.5.3	本章的理论基础和实践内容小结	195
6.6	本章作业与问题	195
6.6.1	本章作业	195
6.6.2	问题：更进一步的思考	195
第 7 章	系统测试与用户验收	196
7.1	集成测试	196
7.1.1	什么是集成测试	196
7.1.2	在“类”级别的集成测试	197
7.1.3	使用 VSTS 的 Web 测试作为集成测试工具	198
7.1.4	用 VSTS 的应用程序设计器创建一个 Web 应用	198
7.1.5	实现 StockBroker 股票交易系统的关键组件	200
7.1.6	配置和创建 Web 测试	204
7.1.7	开始模拟的 Web 测试	205

7.1.8	测试组件之间的操作	206
7.2	系统测试	209
7.2.1	系统功能测试	209
7.2.2	系统性能测试	209
7.2.3	系统负载测试	210
7.3	应用系统的发布过程	212
7.3.1	部署与发布的概念	212
7.3.2	应用系统的现场实施过程	212
7.3.3	应用系统的现场实施活动	213
7.4	模拟用户验收测试	214
7.4.1	制定《验收测试计划》	214
7.4.2	设计《验收测试用例》	215
7.4.3	实施验收测试	216
7.4.4	编写《验收测试报告》	217
7.4.5	模拟用户验收测试	219
7.4.6	修改、修改、再修改	220
7.5	对 ATM 扩展项目进行用户验收测试	220
7.5.1	对 ATM 扩展项目进行验收测试的用户需求	220
7.5.2	对 ATM 扩展项目进行用户验收测试的测试环境	221
7.5.3	对 ATM 扩展项目进行用户验收测试的过程组织	221
7.5.4	对 ATM 扩展项目进行用户验收测试的测试用例	222
7.6	本阶段小结——通过用户验收测试	223
7.6.1	理解实训项目的用户验收测试要求	223
7.6.2	开展实训项目的用户验收测试活动	223
7.6.3	本章的理论基础和实践内容小结	223
7.7	本章作业与问题	223
7.7.1	本章作业	223
7.7.2	问题：更进一步的思考	224
第 8 章	实训项目的结束与总结	225
8.1	理解实训项目的结束活动与要求	225
8.1.1	项目管理中的项目结束活动	225
8.1.2	评价与度量的对象	226
8.1.3	评价“度”的把握	226
8.1.4	评价要素的选择	226
8.1.5	针对项目团队的“评价树”	227
8.1.6	针对实训课程本身的“评价树”	227
8.2	基于项目作品目标实现与否的项目团队评审	228
8.2.1	理解作品目标实现情况的评价与总结要求	228

8.2.2	实训项目作品目标实现情况的测试与评审	228
8.3	基于项目管理目标实现与否的项目团队评审	229
8.3.1	理解项目管理目标实现情况的评价与总结要求	229
8.3.2	项目管理目标实现情况的报告与评审	229
8.4	基于软件过程管理目标实现与否的项目团队评审	230
8.4.1	理解软件过程管理目标实现情况的评价与总结要求	230
8.4.2	实训项目软件过程管理目标实现情况的报告与评审	230
8.5	本阶段小结——对实训课程本身的总结与评价	233
8.5.1	传统课程评价方法的弊端	233
8.5.2	对课程评价方法的改进	234
8.5.3	课程目标与过程评价的实践	235
8.5.4	本章的理论基础和实践内容小结	242
8.6	本章作业与问题	243
8.6.1	本章作业	243
8.6.2	问题：更进一步的思考	243
参考文献		244

软件工程实训项目课程导论

1.1 软件工程与我

1.1.1 角色认知：我的未来不是梦

不论同学们是自觉还是盲目,是喜欢还是厌倦,既然已经选择了软件学院,并且完成了三年的课程学习,即将面临企业实习和走上社会,那么此时同学们的职业去向实际上已经是一个没有多少选择余地的“选答题”。根据统计,软件学院大部分同学毕业后的职业去向,是IT企业的软件工程师。这个职业生涯的发展方向对于软件学院的同学们而言,是最合适的,也是最具有优势的选择。IT企业之外的选择(如公务员、其他行业等)是非常少的,而考研则只是这种选择的“延期”而已,并不算“转行”。这看上去是一种“命运”,这是在你填写高考志愿的时候就已经“确定”了的命运。喜欢的同学自不待言,不喜欢的同学是否有些不服气?

作者本人是恢复高考后第一届计算机软件专业(77级)的学生,在大学的4年时间里,没有见过计算机,更不要说上机编程。只是到了大学毕业以后,分配到计算机厂里,才有每天一小时的上机工作时间。可以想象,在这样的环境下,没有办法培养我对计算机的感情。但是,从业30年,我不但热爱我自己的专业,还把儿子也带进了这个领域。

我想,这里并不需要讨论对自己的专业喜欢还是不喜欢这样的问题。因为喜欢也罢,不喜欢也罢,你实质上已经没有选择地走上了“软件人”这条职业生涯之路。所以,我们所能做的,被动的是适应职业需要,主动的是发现职业生活中的乐趣。有位大师说过,在人生旅途中,如果说谈“理想”可能太沉重了一点,不妨试试把“理想”换成“兴趣”,则可能效果更好一点。“痛苦”地过了一天,“快乐”地过也是过一天。所以,培养你对自己所从事专业的兴趣,从你的工作中发现“乐趣”,可能更有效。这一点,你应该有所体会,当调试程序时,时间过得是那么地快,这是为什么呢?

有人会说,编程有乐趣,但我对软件工程没有兴趣。我说,希望你的乐趣能保持到 50 岁以后……软件工程的乐趣在哪里呢?先看一个例子。

现在,在国内,不但是动车组,甚至高铁都已经开始了商业运行(如图 1-1 所示)。过去,拥挤的人流、从火车的窗子上爬进爬出的画面,在我们的脑子里,还是不太久远的事情。动车怎么保证 2 分钟完成乘客上下车的任务?换一个场景:银行现在普遍采用“电子叫号排队”系统,解决了过去顾客排队的痛苦(如图 1-2 所示)。但是,银行为什么暂时还做不到现场预约时间(拿到号以后,不是告知前面还有多少人在等待,而是告知还需要等待多少时间)。动车对那么大的群体(上下车人流)都能够实现预约到站后的发车时间(2 分钟),银行,前面可能区区只有 10 多个人,为什么做不到时间预约?如果能够实现预约,20 分钟出去吃个早点,而不需要在银行坐等,银行的服务岂不更人性化?

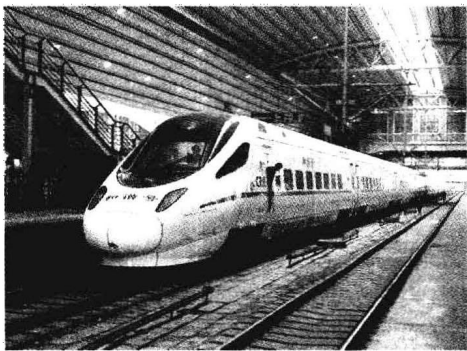


图 1-1 动车的例子



图 1-2 银行排队的例子

为什么动车可以,银行不行呢?过程!动车可以知道在某一站,有多少人上车、下车,他们都在哪节车厢(甚至哪个座位)上,以及他们下车(或上车)最多需要多少时间。所以,一计算——OK! 2 分钟是可以的。银行怎么样呢?不错,你前面可能有三位顾客在等待银行服务,可是,银行完全不知道这三个人到银行来干什么(抢银行也是可能的)?银行怎么告诉你,你还需要等待多久?关键是银行服务的过程,不受银行所控制。

这跟软件工程有什么关系?

看一下图 1-3。对软件工程略有了解的同学,看到这张图一定不会陌生,这是软件过程成熟度 CMM 五级的示意图。对软件开发过程管理的成熟度,通过一个 5 个等级机制进行评价。CMM 对于软件工程的过程而言,是一种过程管理、控制和机制约束,以此保证软件开发结果的水准。因此,现代软件工程更多的是讲软件过程控制和管理。而你的职业生涯在度过了编程这个基本阶段之后,下一个目标就是从事软件开发技术以及工程的控制和管理工作。具体来说,可能是需求分析师、架构师、项目经理等这些更富挑战性、也更具有职业经理人特色的岗位,当然这也同时给你带来与之相应的机遇、挑战、尊重、待遇……你说没有兴趣?

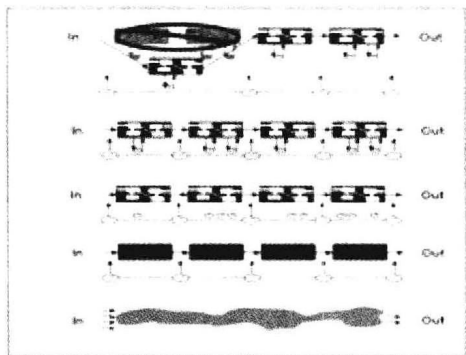


图 1-3 CMM 五级

所以,学习软件工程,是你的个人职业生涯发展中不太遥远、非常真实的需要。当然,更是社会和行业发展的需要,是企业的需要。

按照教育部与原国家计委开办软件学院相关文件的设想,软件学院的培养目标是:为国家培养急需的、符合国内外软件企业发展需求的高层次、应用型、复合型、国际化的软件精英人才。具体而言,软件学院的大部分学生,其未来的职业志向,应该是在未来的若干年后,有望成为国内外大中型软件企业的职业经理人,包括需求分析师、软件构架师、项目经理以及研发部门的技术主管。

在这一点上,我们达成了一致。具体来说,软件学院的培养目标是为软件生产企业培养软件工程的“工程技术”和“管理”人才。而软件工程课程,正是这个专业方向最核心和最关键的课程(必修课程)。

这也说明了软件工程课程所设计的培养对象是什么,培养目标是什么?本课程的(实际上更是本专业的)培养对象和目标是:

- 以现代软件企业、软件开发过程技术管理岗位为对象;
- 以软件项目经理等职业经理人为培养目标。

如果要更进一步形象地描述培养对象和目标的话,不妨把软件企业的开发过程想象为有多个流水线的车间;而培养对象,就是管理这个车间流水线的“拉长”,他们更高目标是成为管理整个车间的车间主任。

在这样的一个软件车间里,车间主任是干什么的?是对目标做出承诺的人,更是控制过程,以实现目标的人。这就是你未来5年,甚至10年的定位。

10年?现在是不是还太早?谁都知道,机会是留给有准备的人。这句话任何时候都是对的。当你学完本课程,知道了一点点软件车间主任应该是什么样子的時候?可以肯定,暂时还没有车间主任给你当,但是,这不妨碍你挑选要去的车间?更不妨碍你挑选老板的时候,拿这个“掂量、掂量”他,这“小子”能教我点什么?别让我的“宝贵青春”,浪费在这个“半吊子”手中吧?你说,这门课“值不值”?

1.1.2 课程定位:软件车间主任的培训班

未来的车间主任要学什么?培训班与“科班”又有什么不同?

1. 教学模式的特色

首先,与一般计算机学院多元化培养、多目标选择不同,软件学院的培养目标是非常具体、明确的,甚至是单一的,就是培养“软件工程”一个专业、一个方向。从某种意义上说,也是非常“功利”的。这个“功”在符合国家、社会发展的需求,而“利”则是对学生个人职业发展有“利”。从某种意义上说,软件学院的“示范性”改革也体现在这里。

在软件工程专业课程群中,“软件工程”是所有专业课程中的核心课程,为了巩固相关课程的知识,进一步强化学生的工程实践能力,“软件工程”课程的实训课程,是对常规课堂教学的一个重要补充和强化,是学生集中一段时间,进行实际软件开发训练的一个重要环节。通过这个阶段的学习和强化训练,同学们能够在具体开发过程中获得现代软件工程知识和实际工程组织、管理能力的极大提升。这是十分必要的,也是软件学院的教学特色。

与常规课堂教学方式不同,在实训课程中,我们有必要的、少量的讲解,最主要的是辅导