

<<软件工程项目实训教程>>

图书基本信息

书名：<<软件工程项目实训教程>>

13位ISBN编号：9787302248958

10位ISBN编号：7302248958

出版时间：2011-6

出版时间：清华大学出版社

作者：张家浩 编

页数：243

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<软件工程项目实训教程>>

### 内容概要

《软件工程项目实训教程：基于微软vsts》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。

《软件工程项目实训教程：基于微软vsts》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。

## &lt;&lt;软件工程项目实训教程&gt;&gt;

## 书籍目录

## 第1章 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

## 1.1.2 课程定位：软件车间主任的培训班

## 1.1.3 实训与实战：态度决定一切

## 1.1.4 理论与实践：相映成趣、相得益彰

## 1.2 实训各阶段的任务与要求

## 1.2.1 阶段1：组成项目团队

## 1.2.2 阶段2：提出项目目标

## 1.2.3 阶段3：制定项目计划

## 1.2.4 阶段4：完成需求定义

## 1.2.5 阶段5：设计系统架构

## 1.2.6 阶段6：完成代码开发

## 1.2.7 阶段7：提交测试验收

## 1.2.8 阶段8：进行项目总结

## 1.3 实训项目课题的选择

## 1.3.1 关于实训项目课题的选择

## 1.3.2 实训项目案例——atm系统扩展

## 1.3.3 在相同基础上的个性化发挥

## 1.4 带着兄弟们上路

## 1.4.1 组成项目团队

## 1.4.2 从角色需要考虑人选

## 1.4.3 准备vsts环境

## 1.4.4 阅读教材与参考书

## 1.5 本阶段小结——组成项目团队

## 1.5.1 理解实训项目的课程目标

## 1.5.2 了解如何进行课程阶段成果的检查

## 1.5.3 本章的理论基础和实践内容小结

## 1.6 本章作业与问题

## 1.6.1 本章作业

## 1.6.2 问题：更进一步的思考

## 第2章 实训项目的选题与目标范围确定

## 2.1 认识我们的目标

## 2.1.1 微软创新杯

## 2.1.2 大赛主题与选题

## 2.1.3 理解大赛主题

## 2.2 有一个好的点子

## 2.2.1 基于传统问题域的创新选题

## 2.2.2 关注新的应用领域

## 2.2.3 寻找新的技术应用热点

## 2.2.4 参考各类软件创新大赛的获奖选题

## 2.2.5 从既往的项目中选择课题

## 2.3 选题的目标设计与价值评价

## 2.3.1 目标与方案

## 2.3.2 最初的目标：大赛通知

## <<软件工程项目实训教程>>

- 2.3.3 第一个交付成果：《项目计划书》
- 2.3.4 再谈目标——大赛的评价标准
- 2.4 课题的难度与可行性
  - 2.4.1 何谓“难度”
  - 2.4.2 难度与可行性的关系
  - 2.4.3 可行性的非技术性考虑
- 2.5 确定项目的范围
  - 2.5.1 为目标而确定范围
  - 2.5.2 考虑条件和资源
  - 2.5.3 修剪你的“范围树”
- 2.6 项目团队组建与项目初步规划
  - 2.6.1 角色与分工
  - 2.6.2 项目总体规划
  - 2.6.3 为第一次提交，制订更详细的工作计划
- 2.7 实训项目案例——《atm扩展项目计划书》
  - 2.7.1 参赛作品构思的创意与价值
  - 2.7.2 参赛作品的目标实现形式
  - 2.7.3 参赛作品目标实现的可行性
  - 2.7.4 团队组成与角色分工
  - 2.7.5 项目时间进度表
- 2.8 本阶段小结——通过项目初审
  - 2.8.1 软件创新大赛各阶段评审重点
  - 2.8.2 第一轮入围与评判标准
  - 2.8.3 本章的理论基础和实践内容小结
- 2.9 本章作业与问题
  - 2.9.1 本章作业
  - 2.9.2 问题：更进一步的思考
- 第3章 交付过程模型与项目管理控制
  - 3.1 交付过程模型与过程管理
    - 3.1.1 过程模型的一般意义
    - 3.1.2 微软公司的软件过程模型msf与vsts
    - 3.1.3 统一过程模型rup与ibm的jazz
    - 3.1.4 msf与rup的比较
  - 3.2 交付过程模型的结构与关键行为
    - 3.2.1 工作项与工作产品
    - 3.2.2 角色
    - 3.2.3 流程
    - 3.2.4 活动和步骤
  - 3.3 为实训项目搭建vsts平台
    - 3.3.1 vsts的逻辑结构与物理结构
    - 3.3.2 安装vsts系统的服务器端
    - 3.3.3 安装vsts系统的客户端
  - 3.4 使用vsts定义项目
    - 3.4.1 在vsts上定义项目
    - 3.4.2 使用团队资源管理器定义项目的工作项内容
  - 3.5 为项目定义基线与状态
    - 3.5.1 基线与状态控制

## &lt;&lt;软件工程项目实训教程&gt;&gt;

- 3.5.2 用vsts设置项目的基线
- 3.6 创建门户与团队报告
  - 3.6.1 打开团队项目门户
  - 3.6.2 自定义并扩展团队报告
- 3.7 实训项目案例——atm扩展
  - 3.7.1 定义前的准备
  - 3.7.2 为atm扩展项目选择生命周期模型
  - 3.7.3 为atm扩展项目定义项目计划
  - 3.7.4 为atm扩展项目定义具体的工作项内容
  - 3.7.5 为atm扩展项目定义基线
  - 3.7.6 查看atm扩展项目的团队门户和团队报告
- 3.8 本阶段小结——召开第一次项目例会
  - 3.8.1 如何开好一次项目例会
  - 3.8.2 本章的理论基础和实践内容小结
- 3.9 本章作业与问题
  - 3.9.1 本章作业
  - 3.9.2 问题：更进一步的思考
- 第4章 需求工程中的需求开发与管理
  - 4.1 软件需求的获取与描述
    - 4.1.1 需求获取阶段的工作目标与关键交付物成果
    - 4.1.2 基于uml的电梯控制系统需求模型
    - 4.1.3 项目案例：atm扩展项目的需求获取过程
    - 4.1.4 项目案例：atm基本系统的业务用例模型
    - 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
    - 4.1.6 项目案例：atm系统的扩展以及相关的用户确认
  - 4.2 需求分析模型与关键需求
    - 4.2.1 需求分析阶段的工作目标与关键交付物成果
    - 4.2.2 项目案例：atm基本系统的需求分析模型
    - 4.2.3 项目案例：atm扩展的需求场景
    - 4.2.4 项目案例：atm扩展的关键需求分析
    - 4.2.5 项目案例：atm关键需求实现的用户验收
  - 4.3 文档化的需求处理与需求规格描述
    - 4.3.1 需求处理阶段的工作目标与关键交付物成果
    - 4.3.2 《需求规格说明书》的主要内容
    - 4.3.3 项目案例：atm项目的《需求规格说明书》
  - 4.4 borlandcaliber的需求定义与管理功能
    - 4.4.1 borlandcaliber的需求定义与管理概念
    - 4.4.2 borlandcaliber的需求定义与管理过程
  - 4.5 使用caliber定义项目需求
    - 4.5.1 创建新项目
    - 4.5.2 为项目创建需求树
    - 4.5.3 定义需求信息
    - 4.5.4 为需求分配属性值
    - 4.5.5 为需求分配用户和组
    - 4.5.6 链接参考文档
    - 4.5.7 创建追踪能力链接
    - 4.5.8 定义需求验证过程

## &lt;&lt;软件工程项目实训教程&gt;&gt;

- 4.6 使用caliber对需求进行基线管理
  - 4.6.1 查看项目的需求基线
  - 4.6.2 创建项目的需求基线
  - 4.6.3 初始化项目的需求基线
  - 4.6.4 锁定项目的需求基线
  - 4.6.5 项目基线的电子签名
  - 4.6.6 添加项目基线的电子签名
  - 4.6.7 查看项目基线的签名报告
  - 4.6.8 比较项目的需求基线
  - 4.6.9 有关项目需求基线的小结
- 4.7 将caliber与vsts集成
  - 4.7.1 安装caliber的vsts插件
  - 4.7.2 让caliber与vsts工作项关联
  - 4.7.3 在vsts中创建caliber的需求项
  - 4.7.4 建立caliber的需求项与vsts工作项之间的跟踪关系
  - 4.7.5 总结：caliber与vsts一起工作
- 4.8 在caliber上建立atm扩展需求
  - 4.8.1 从业务用例模型到caliber需求树
  - 4.8.2 了解caliber上的atm需求
  - 4.8.3 在caliber中扩展自己的atm需求
  - 4.8.4 在caliber上确定项目的需求范围边界
  - 4.8.5 与vsts工作项相关联并确定基线
- 4.9 本阶段小结——通过需求评审
  - 4.9.1 理解实训项目的需求评审要求
  - 4.9.2 开展实训项目的需求评审活动
  - 4.9.3 本章 的理论基础和 实践内容小结
- 4.10 本章 作业与问题
  - 4.10.1 本章 作业
  - 4.10.2 问题：更进一步的思考
- 第5章 基于关键质量属性的架构设计
  - 5.1 最初的架构模型设想
    - 5.1.1 从需求模型开始
    - 5.1.2 一般架构模型的基本考虑
    - 5.1.3 从需求模型到架构模型的转换
    - 5.1.4 电梯控制系统的架构设计
  - 5.2 关注与架构有关的关键质量属性
    - 5.2.1 满足关键质量属性需求的架构设计
    - 5.2.2 电梯控制系统关键质量属性需求分析
    - 5.2.3 规范的关键质量属性场景描述
    - 5.2.4 有关atm扩展的关键质量属性场景描述
  - 5.3 基于关键质量属性需求的分析与架构设计
    - 5.3.1 “可用性”需求的现状分析
    - 5.3.2 实现“实时性”需求的对策
    - 5.3.3 实现“实时性”需求的方法
    - 5.3.4 基于关键质量属性的架构分析
    - 5.3.5 实时故障恢复系统的架构设计考虑
    - 5.3.6 实时故障恢复系统的详细设计考虑

## &lt;&lt;软件工程项目实训教程&gt;&gt;

- 5.4 搭建一个基于mvc模式的struts架构
  - 5.4.1 选择mvc模式的理由
  - 5.4.2 用struts搭建一个“轻量级”的应用架构
  - 5.4.3 更进一步地体验struts架构中的mvc组件
  - 5.4.4 项目作业：比较在struts架构上搭建atm系统的优劣
- 5.5 在struts架构上运用面向对象设计模式
  - 5.5.1 业务处理流程灵活性的质量属性场景描述
  - 5.5.2 实现业务处理流程灵活性的战术对策
  - 5.5.3 采用工厂方法实现流程灵活性的关键质量需求
  - 5.5.4 在struts框架下实现设计模式的应用
- 5.6 使用vsts可视化的分布式系统设计器构建系统架构
  - 5.6.1 分布式系统设计器的作用和相互关系
  - 5.6.2 定义组件的提供者
  - 5.6.3 定义对组件提供者终结点的控制
  - 5.6.4 定义组件之间的连接
  - 5.6.5 应用程序的实现
  - 5.6.6 项目作业：使用vsts应用程序设计器实现atm系统
- 5.7 架构文档与架构评审
  - 5.7.1 规范的架构设计活动过程与制品
  - 5.7.2 需要编写的架构视图和文档
  - 5.7.3 透过架构视图表现架构设计的核心内容
  - 5.7.4 针对一般要素的架构设计评审
  - 5.7.5 针对关键质量属性需求的架构设计评审
  - 5.7.6 atm实时故障恢复系统的构架设计评审
- 5.8 本阶段小结——通过架构设计评审
  - 5.8.1 理解项目实训课程的架构设计评审要求
  - 5.8.2 开展实训项目的架构评审活动
  - 5.8.3 本章的理论基础和实践内容小结
- 5.9 本章作业与问题
  - 5.9.1 本章作业
  - 5.9.2 问题：更进一步的思考
- 第6章 代码开发阶段的软件过程控制与管理
  - 6.1 用vsts实现对源代码的控制与管理
    - 6.1.1 从建立规范的源代码开发管理环境开始
    - 6.1.2 使用源代码管理器对个人的工作区进行管理
    - 6.1.3 向存储库和工作区中添加文件夹/文件/解决方案
    - 6.1.4 通过配置签入/签出策略设置对变更活动的约束
    - 6.1.5 签出
    - 6.1.6 签入
  - 6.2 使用tfvc进行版本控制
    - 6.2.1 设置团队版本控制环境
    - 6.2.2 决定控制什么和由谁来进行控制
    - 6.2.3 在开发中使用tfvc进行版本控制和管理
  - 6.3 使用tfb进行构建与发布管理
    - 6.3.1 什么是现代构建与发布管理
    - 6.3.2 安装并配置tfb
    - 6.3.3 使用tfb进行生成与每日集成

## &lt;&lt;软件工程项目实训教程&gt;&gt;

- 6.3.4 查看tfb报告
- 6.4 体验vsts的单元测试与测试管理
  - 6.4.1 测试的概念与vsts的测试功能
  - 6.4.2 单元测试的概念
  - 6.4.3 建立本地的单元测试环境
  - 6.4.4 为单元测试设置vs项目
  - 6.4.5 使用测试管理器运行和管理单元测试
  - 6.4.6 尝试vsts的测试代码覆盖
  - 6.4.7 运用vsts托管代码分析工具
  - 6.4.8 将测试与tfs集成并发布测试结果
- 6.5 本阶段小结——通过代码评审
  - 6.5.1 理解实训项目的代码评审要求
  - 6.5.2 开展实训项目的代码评审活动
  - 6.5.3 本章的理论基础和实践内容小结
- 6.6 本章 作业与问题
  - 6.6.1 本章 作业
  - 6.6.2 问题：更进一步的思考
- 第7章 系统测试与用户验收
  - 7.1 集成测试
    - 7.1.1 什么是集成测试
    - 7.1.2 在“类”级别的集成测试
    - 7.1.3 使用vsts的web测试作为集成测试工具
    - 7.1.4 用vsts的应用程序设计器创建一个web应用
    - 7.1.5 实现stockbroker股票交易系统的关键组件
    - 7.1.6 配置和创建web测试
    - 7.1.7 开始模拟的web测试
    - 7.1.8 测试组件之间的操作
  - 7.2 系统测试
    - 7.2.1 系统功能测试
    - 7.2.2 系统性能测试
    - 7.2.3 系统负载测试
  - 7.3 应用系统的发布过程
    - 7.3.1 部署与发布的概念
    - 7.3.2 应用系统的现场实施过程
    - 7.3.3 应用系统的现场实施活动
  - 7.4 模拟用户验收测试
    - 7.4.1 制定《验收测试计划》
    - 7.4.2 设计《验收测试用例》
    - 7.4.3 实施验收测试
    - 7.4.4 编写《验收测试报告》
    - 7.4.5 模拟用户验收测试
    - 7.4.6 修改、修改、再修改
  - 7.5 对atm扩展项目进行用户验收测试
    - 7.5.1 对atm扩展项目进行验收测试的用户需求
    - 7.5.2 对atm扩展项目进行用户验收测试的测试环境
    - 7.5.3 对atm扩展项目进行用户验收测试的过程组织
    - 7.5.4 对atm扩展项目进行用户验收测试的测试用例



## <<软件工程项目实训教程>>

7.6 本阶段小结——通过用户验收测试

7.6.1 理解实训项目的用户验收测试要求

7.6.2 开展实训项目的用户验收测试活动

7.6.3 本章的理论基础和实践内容小结

7.7 本章作业与问题

7.7.1 本章作业

7.7.2 问题：更进一步的思考

第8章 实训项目的结束与总结

8.1 理解实训项目的结束活动与要求

8.1.1 项目管理中的项目结束活动

8.1.2 评价与度量的对象

8.1.3 评价“度”的把握

8.1.4 评价要素的选择

8.1.5 针对项目团队的“评价树”

8.1.6 针对实训课程本身的“评价树”

8.2 基于项目作品目标实现与否的项目团队评审

8.2.1 理解作品目标实现情况的评价与总结要求

8.2.2 实训项目作品目标实现情况的测试与评审

8.3 基于项目管理目标实现与否的项目团队评审

8.3.1 理解项目管理目标实现情况的评价与总结要求

8.3.2 项目管理目标实现情况的报告与评审

8.4 基于软件过程管理目标实现与否的项目团队评审

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

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

8.5 本阶段小结——对实训课程本身的总结与评价

8.5.1 传统课程评价方法的弊端

8.5.2 对课程评价方法的改进

8.5.3 课程目标与过程评价的实践

8.5.4 本章的理论基础和实践内容小结

8.6 本章作业与问题

8.6.1 本章作业

8.6.2 问题：更进一步的思考

参考文献

## &lt;&lt;软件工程项目实训教程&gt;&gt;

## 章节摘录

版权页：插图：首先看到，作为初赛入围的评价标准，初赛阶段评委关注什么？

在初赛阶段，评委首先关注的是参赛项目的创意与实用性，这个部分占了初赛成绩的40%。

而在大赛总的评价标准（表2-1）中，问题定义和解决方案设计 and 创新两个部分，对此做了比较明确的提示。

这个总体要求，主要体现在《项目计划书》的第1节“系统主题”和第2节“需求分析”中。

系统主题是《项目计划书》中最核心、最关键的内容。

如果你很难揣摩评委的评价标准的话，我们试试从反面看看评委们从提交的文档中能看出什么？

评委如何从文档中进行筛选？

而不被评委淘汰就是我们的目标！

换位思考一下：评委从几千份《项目计划书》中看什么（能看出什么）？

评委不看什么（看不出什么）？

如果你是评委，拿到这10页纸（超过10页，评委一定不会给你加分）的文档，一般会先看一下《项目计划书》是否按目录的要求去写。

这主要是先筛掉一些不按规则出牌的人。

第一轮可能有成千上万的队伍，评委们没有时间和兴趣与那些“乱来”的人“过招”。

“形式”没有问题，那好，顺着目录顺序（关键是顺着评委自己的思路），看几个关键的、核心的内容是否有。

什么是关键问题？

参赛作品到底要干什么（系统主题），有什么价值（如何体现创新性和应用价值）？

这个问题是核心关键，是第一轮淘汰目标。

在这个关键问题上首先会犯的错误是：你自己都不知道自己到底要干什么（这个问题我们在后面将详细讲解）？

你都不知道做出来的是个什么东西，你怎么让评委知道呢？

其次才是这个东西有什么价值。

什么不是关键问题？

用什么技术、平台、实现的功能是否完整（具有核心价值的功能除外），文档是否符合软件工程规范（现在不看）等。

所以，在第一阶段所提交的《项目计划书》中，目标是：严格按模板编写《项目计划书》，在规定的格式下，用最简单、最清晰的方法，描述参赛作品的最终的呈现形式和结果（创新功能），用最能打动评委的语言，描述作品结果的价值。

如果时间来不及，“系统主题”部分要花80%的时间和精力，认真准备，其他内容可以简单一点，不要花过多的时间和精力，更不要因此而影响“系统主题”的编写。

## <<软件工程项目实训教程>>

### 编辑推荐

《软件工程项目实训教程:基于微软VSTS》是重点大学软件工程规划系列教材之一。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>