

<<软件工程导论>>

图书基本信息

书名：<<软件工程导论>>

13位ISBN编号：9787302272519

10位ISBN编号：7302272514

出版时间：2012-1

出版时间：清华大学出版社

作者：(美)夏洛特 著, 罗飞 等译

页数：198

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>

<<软件工程导论>>

内容概要

《软件工程导论》的安排方式很简单，它的每一章对应项目开发中的一个主要活动。工程学方面的主要活动包括需求分析和需求规范、体系结构设计、模块设计、编码和单元测试、以及测试；项目管理方面的主要活动包括项目计划和项目监控两个方面，而这两个活动都安排在项目计划这一章，这是因为项目的监控也需要计划。另外，在这本书中，还有一章主要用来阐明软件工程领域存在的问题，而另一章则讨论了软件过程中把每个活动联系起来的中心观点。

《软件工程导论》的每一章都是以对本章的介绍、目标或者读者在本章中应有的收获开始。每一章在介绍项目的开发活动时，总是首先介绍有关概念和知识，然后介绍该活动的结果或所具备的某些期望质量特性，以及一些实践方法或执行该活动所需要的技术，最后通过一些实例说明这一章的有关知识点，并为读者总结本章学习的主要内容，每一章的结尾都提供有一些自测练习。

<<软件工程导论>>

作者简介

作者:(美)夏洛特 译者:罗飞、邵凌霜

<<软件工程导论>>

书籍目录

第1章 软件问题

- 1.1 成本、进度和质量
- 1.2 规模和变更
- 1.3 小结

自测练习

第2章 软件过程

- 2.1 过程和项目
- 2.2 软件过程的组成
- 2.3 软件开发过程模型
 - 2.3.1 瀑布模型
 - 2.3.2 原型模型
 - 2.3.3 迭代开发模型
 - 2.3.4 rational统一过程模型
 - 2.3.5 时间盒模型
 - 2.3.6 极限编程 (xp) 和敏感过程模型
 - 2.3.7 过程模型在项目中的应用
- 2.4 项目管理过程
- 2.5 小结

自测练习

第3章 软件需求分析和软件需求规格

- 3.1 好软件需求规格的意义
- 3.2 需求过程
- 3.3 需求规格
 - 3.3.1 软件需求规格应该具备的特点
 - 3.3.2 软件需求规格的组成
 - 3.3.3 需求文档的结构
- 3.4 用例驱动功能规格
 - 3.4.1 基础知识
 - 3.4.2 几个例子
 - 3.4.3 扩展
 - 3.4.4 构建用例
- 3.5 其他分析方法
 - 3.5.1 数据流图
 - 3.5.2 er图
- 3.6 验证
- 3.7 小结

自测练习

第4章 软件计划

- 4.1 工作量估算
 - 4.1.1 自顶向下估算方法
 - 4.1.2 自底向上估算方法
- 4.2 项目进度和人员配备
- 4.3 质量计划
- 4.4 风险管理计划
 - 4.4.1 风险管理的观念

<<软件工程导论>>

- 4.4.2 风险评估
- 4.4.3 风险控制
- 4.4.4 一个实用的风险管理计划方法
- 4.5 项目监测计划
- 4.5.1 项目的度量
- 4.5.2 项目监测和跟踪
- 4.6 详细日程安排
- 4.7 小结
- 自测练习

第5章 软件体系结构

- 5.1 软件体系结构的作用
- 5.2 体系结构视图
- 5.3 构件和连接件视图
- 5.3.1 构件
- 5.3.2 连接件
- 5.3.3 举例
- 5.4 构件和连接件视图的体系结构模式
- 5.4.1 管道-过滤器模式
- 5.4.2 共享数据模式
- 5.4.3 客户端-服务器模式
- 5.4.4 其他模式
- 5.5 体系结构设计的文档化
- 5.6 体系结构评估
- 5.7 小结
- 自测练习

第6章 设计

- 6.1 设计的基本概念
- 6.1.1 耦合
- 6.1.2 内聚
- 6.1.3 开闭原则
- 6.2 面向功能设计
- 6.2.1 结构图
- 6.2.2 结构化设计方法
- 6.2.3 举例
- 6.3 面向对象设计
- 6.3.1 面向对象基本概念
- 6.3.2 统一建模语言uml
- 6.3.3 设计方法论
- 6.3.4 举例
- 6.4 详细设计
- 6.4.1 逻辑/算法设计
- 6.4.2 类状态模型
- 6.5 验证
- 6.6 复杂性度量
- 6.6.1 面向功能设计的复杂性度量
- 6.6.2 面向对象设计的复杂性度量
- 6.7 小结

<<软件工程导论>>

自测练习

第7章 编码和单元测试

- 7.1 编程原则和指南
 - 7.1.1 结构化编程
 - 7.1.2 信息隐藏
 - 7.1.3 程序设计实践经验
 - 7.1.4 编码标准
- 7.2 增量开发
 - 7.2.1 一个增量编码方法
 - 7.2.2 测试驱动开发
 - 7.2.3 结对编程
- 7.3 代码演化的管理
 - 7.3.1 源代码控制和生成
 - 7.3.2 重构
- 7.4 单元测试
 - 7.4.1 程序过程单元测试
 - 7.4.2 类单元测试
- 7.5 代码检查
 - 7.5.1 计划
 - 7.5.2 代码自查
 - 7.5.3 小组会议评审
- 7.6 代码度量
 - 7.6.1 代码规模测量
 - 7.6.2 复杂性度量
- 7.7 小结

自测练习

第8章 测试

- 8.1 测试概念
 - 8.1.1 错误、缺陷和失败
 - 8.1.2 测试用例、测试集和测试配置
 - 8.1.3 测试心理
 - 8.1.4 测试层次
- 8.2 测试过程
 - 8.2.1 测试计划
 - 8.2.2 测试用例设计
 - 8.2.3 测试用例执行
- 8.3 黑盒测试
 - 8.3.1 等价类划分
 - 8.3.2 边界值分析
 - 8.3.3 成对测试
 - 8.3.4 特殊情况
 - 8.3.5 基于状态的测试
- 8.4 白盒测试
 - 8.4.1 基于控制流的测试准则
 - 8.4.2 测试用例生成及支持工具
- 8.5 度量标准
 - 8.5.1 覆盖率分析

<<软件工程导论>>

8.5.2 可靠性

8.5.3 缺陷消除率

8.6 小结

自测练习

参考文献

对照表

章节摘录

版权页：插图：3.4.1 基础知识一个软件系统可能被许多用户或其他系统使用，我们用于揭示需求。

在用例术语中角色是一个人或系统，它用系统而达到一个目标。

注意角色与系统交互而达到一些目标，那么角色是逻辑实体，它代表着一群有相同行为的用户（人或系统）。

不同的角色因为不同的目标代表着不同的群体，所以对系统发送某些消息而由另外用户接受消息的普通“使用者”，采用“接受者”和“发送者”来称谓比较好。

主角色是使用用例来达到某个目标的主要角色，而满足这一目的是该用例的主要目标。

主角色是逻辑概念，尽管我们假设主角色执行用例，但是一些代理可能会真正执行主角色用例中的行为。

例如，用区域用例获得销售增长报告的VP可以是主角色，尽管真正执行的是秘书。

我们把主角色认为是那个真正使用用例结果的人，以及目标中的主要消费者。

时间驱动触发器是关于主角色怎样执行用例（在这种情形下，有时报告是自动生成的）的另一个例子。

注意，虽然主角色的目标是用例的驱动力，但是用例也需要满足其他相关者的目标。

那就是说，尽管用例可由主角色的目标来驱动，但是用例的目的是描述系统满足所有用户的目标行为。

例如，一个从ATM机上取款的用例，它的客户是主角色，将正常地描述客户与AIM之间的全部交互。

然而，银行也是AIM系统相关者，它的目标包括记录所有的步骤、当账户里有足够资金时钱才被取出、一次取款不能超过多少限额等。

ATM取款用例中要满足所有这些目标。

为了描述交互行为，用例采用情景。

情景描述了在某些特定条件下为达到某一目标而执行的活动集。

活动集通常描述为明确的顺序，尽管有些操作的执行以并行或其他次序，情景中的每一步都是由系统或角色执行的一个逻辑上完整的活动，通常是角色的某个行为，每一逻辑步骤是系统为达到某目标的向前发展，或者是为了满足某些目标而改变一些中间状态。

<<软件工程导论>>

编辑推荐

《软件工程导论》是计算机科学本科核心课程教材之一。

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>