

# Week 11

Nite

2025-05-08T17:18:00+11:00

阅读 PDF

## 复习重点

### 软件质量工程工作计划

- **资源:**

- 附录 A1: 什么是软件质量? (What is Software Quality?)
- 附录 A2: 人员能力成熟度模型 (People Capability Maturity Model, P-CMM)

- **工作计划:**

1. **软件质量 (Software Quality)** 的定义。

- 软件质量是指产品满足既定需求的程度, 这些需求准确反映了利益相关者的需要、期望和愿望。

Software quality refers to the degree to which a product meets specified requirements, which accurately reflect the needs, expectations, and desires of stakeholders.

2. 软件质量工程 (Software Quality Engineering, SQE) 的目的、任务、机制、目标和手段。

- 目的:

1. 满足需求 (Meeting the Requirements)
2. 实现利益相关者满意度 (Producing Stakeholder Satisfaction)

- 任务:

1. Quality Planning
2. Requirements Analysis
3. Design for Quality
4. Code Reviews and Inspections
5. Testing Activities
6. Defect Management
7. Process Improvement
8. Metrics and Measurement
9. Configuration and Change Management
10. Customer Satisfaction and Feedback Analysis

- 机制:

1. Quality Assurance(QA)
2. Quality Control(QC)

3. Static Analysis Tools
  4. Dynamic Testing Tools
  5. Test Automation
  6. Continuous Integration/Continuous Delivery(CI/CD)
- 目标:
    1. Prevent Defects
    2. Ensure Compliance
    3. Improve Efficiency
    4. Enhance User Experience
    5. Increase Reliability and Availability
  - 手段:
    1. Standards(IEEE 730-2014)
    2. Methodologies
    3. Tools
    4. Training
    5. Audits
    6. Feedback Loops
3. 软件开发中**质量规划 (Quality Planning)**、**质量保证 (Quality Assurance)** 和**质量控制 (Quality Control)** 的区别。
- **质量规划**: 制定质量目标和实现计划, 确定质量标准和流程 (Develop quality objectives and achievement plans, define quality standards and processes)。
  - **质量保证**: 通过系统性活动 (如审计、审查) 确保开发过程符合质量标准 (Ensure that the development process meets quality standards through systematic activities (e.g., audits, reviews))。
  - **质量控制**: 检查具体输出 (如代码、产品), 识别和纠正缺陷 (Inspect specific outputs (e.g., code, products) to identify and correct defects)。
4. **软件需求规格说明书 (Software Requirement Specification, SRS)** 的定义、用途和目的。
- 定义: 一份明确定义正在开发的系统的文档 (a document that clearly define the system under development)。
  - 用途:
    1. 沟通工具: 作为客户、用户、开发人员、测试人员、项目经理之间沟通的基础。  
Communication tools: as the basis for communication between customers, users, developers, testers, and project managers.
    2. 设计依据: 为后续的设计、编码、测试提供明确的依据。  
Design basis: provides a clear basis for subsequent design, coding, and testing.
    3. 验收标准: 作为验证和确认软件是否满足用户需求的标准。  
Acceptance criteria: as a standard for verifying and confirming whether the software meets user requirements.
    4. 项目管理基础: 用于估算工作量、制定进度计划、分配资源。

Project management foundation: used to estimate workload, develop schedules, and allocate resources.

5. 变更控制参考：当需求发生变化时，SRS 是判断变更影响的重要依据。

Change control reference: when requirements change, SRS is an important basis for judging the impact of changes.

- 目的：

- \* 理解并明确需求 (To understand and specify requirements)
- \* 提供系统功能的精确说明 (To provide precise statement about the functions of the system)
- \* 为成本和进度估算提供基础 (To provide basis for costs and schedules estimations)
- \* 提供验证和确认的基准 (To provide a baseline for validation and verification)
- \* 缓解语言障碍问题 (To mitigate problem of language barrier)
- \* 减轻理解和阅读文档的困难 (Mitigating difficulties in understanding and reading documents)
- \* 防止需求混淆 (To prevent requirements confusion)
- \* 防止功能需求和非功能需求的混杂 (To prevent requirements confusion)
- \* 防止需求组合 (To prevent requirements combination)
- \* 减少开发工作量 (To reduce the development efforts)
- \* 作为增强的基础 (To serve as a basis for enhancement)

## 5. 人员能力成熟度模型 (P-CMM) 的定义、阶段及其在软件开发环境中的相关性。

- **定义：** P-CMM 是一种用于评估和改进组织人力资源能力成熟度的框架，旨在通过系统化的人力资本管理提升员工绩效与组织竞争力。

**P-CMM (People Capability Maturity Model)** is a framework used to assess and improve the maturity of an organization's human resource capabilities, aiming to enhance employee performance and organizational competitiveness through systematic human capital management.

- **阶段：**

1. **初始级 (Initial)：** 人力资源实践是随意的、反应式的，缺乏标准化流程。依赖个别员工的能力而非系统性方法。
2. **可重复级 (Repeatable)：** 建立了基本的人力资源政策和流程（如招聘、入职、绩效评估），可以重复使用并适应类似情境。
3. **定义级 (Defined)：** 人力资源实践被制度化、标准化，并与组织的战略目标相结合，形成统一的流程指南和角色职责。
4. **已管理级 (Managed)：** 使用量化指标对人力资源流程进行测量和控制，确保其有效性和一致性。
5. **优化级 (Optimizing)：** 组织持续改进人力资源流程，通过创新和最佳实践推动员工能力和组织绩效的不断提升。

- **在软件开发中的应用：**

- \* 提升团队效能 (Improve team effectiveness)
- \* 支持过程改进 (Support process improvement)

- \* 降低项目风险 (Reduce project risks)
  - \* 增强组织竞争力 (Enhance organisational competitiveness)
  - \* 促进知识管理和持续学习 (Promote knowledge management and continuous learning)
6. 假设某竞争公司要求“代码审查 (Code Inspections) 必须在至少两个层级的层级结构中进行”，在软件质量工程 (SQE) 背景下，这意味着什么？
- 这通常意味着该公司希望将代码审查过程系统化、结构化，并通过多层审核机制来提高代码质量和缺陷发现率。

This usually means that the company wants to systematise and structure the code review process and improve code quality and defect detection rates through a multi-layered review mechanism.

### 软件质量规划工作计划

- **资源:**

- 附录 B: IEEE 730-2014 和 ISO/IEC 25010:2023 标准

- **工作计划:**

- 阅读资源文档，回答以下问题并注明来源：

1. **IEEE 730-2014 标准** 的目的。

- \* 本标准的目的是为软件项目提供统一的、最低可接受的 SQA 流程要求。

The purpose of this standard is to provide uniform, minimum acceptable requirements for SQA processes in support of a software project.

2. 选择 IEEE 730-2014 建议的一个文档并描述其内容。

- \* TODO

3. 选择一个适用于你产品的 IEEE 730-2014 **审查流程 (Review Processes)** 并描述其细节 (不超过三段)。

- \* TODO

4. 列出产品中与 ISO/IEC 25010:2023 标准相关的五个特性，涉及时间行为 (Time Behavior)、资源利用 (Resource Utilization)、互操作性 (Interoperability)、可学习性 (Learnability)、用户错误保护 (User Error Protection)、容错性 (Fault Tolerance)、可维护性 (Maintainability)。

- \* TODO

### 用例描述与图表示例

- **资源:**

- 附录 G: 用例笔记 (Use Case Notes)

- **场景:**

- 物流部门有一个支持客户请求和库存水平 (Stock Levels) 的应用程序。客户提交请求，客户代表处理订单，订单准备好后通知客户付款。如果某些物品的库存达到特定水平，代表可以选择重新订货。

• **工作计划:**

- 以团队形式描述应用程序界面：
  1. 为其中一个步骤编写**用例 (Use Case)**。

字段 (Field)	内容 (Content)
<b>用例名称 (Use Case)</b>	客户下订单 (Place Customer Order)
<b>目标 (Goal)</b>	客户成功下订单并完成支付流程，同时系统检查库存是否需要补货。
<b>范围与级别 (Scope &amp; Level)</b>	物流管理系统，用户级别 (User-goal level)
<b>前置条件 (Pre-condition)</b>	系统已启动，客户账户可用，库存信息同步完成。
<b>成功结束条件 (Success End Condition)</b>	订单成功记录，客户完成支付，若库存不足则生成补货通知。
<b>失败结束条件 (Failed End Condition)</b>	客户未完成支付或系统库存检查失败，订单未被确认。
<b>参与者 (Actors)</b>	客户 (Customer)、客户代表 (Customer Representative)
<b>触发器 (Trigger)</b>	客户提交订单请求。
<b>描述 (Description)</b>	1. 客户登录系统并浏览产品。2. 客户选择商品并下订单。3. 系统记录订单并通知客户代表。4. 客户代表检查库存。5. 若库存充足，系统准备发货并通知客户支付。6. 客户完成支付。7. 若库存不足，系统提示客户代表是否下补货单。8. 系统更新库存信息。
<b>扩展 (Extends)</b>	支付失败流程 (用于描述支付失败时的处理)。
<b>包含 (Includes)</b>	检查库存 (Check Stock)、生成补货单 (Create Re-stocking Order)
<b>子变体 (Sub Variations)</b>	客户批量下单、客户预约未来发货
<b>利益相关者及利益 (Stakeholders and Interests)</b>	客户希望及时收到订单；公司希望避免缺货；仓库希望优化库存。
<b>使用频率 (Frequency of Use)</b>	高频 (每天多次)
<b>风险级别 (Level of Risk)</b>	中等，涉及支付和库存。

字段 (Field)	内容 (Content)
优先级 (Priority)	高

2. 为整个场景绘制用例描述 (Use Case Description) 和用例图 (Use Case Diagram)。

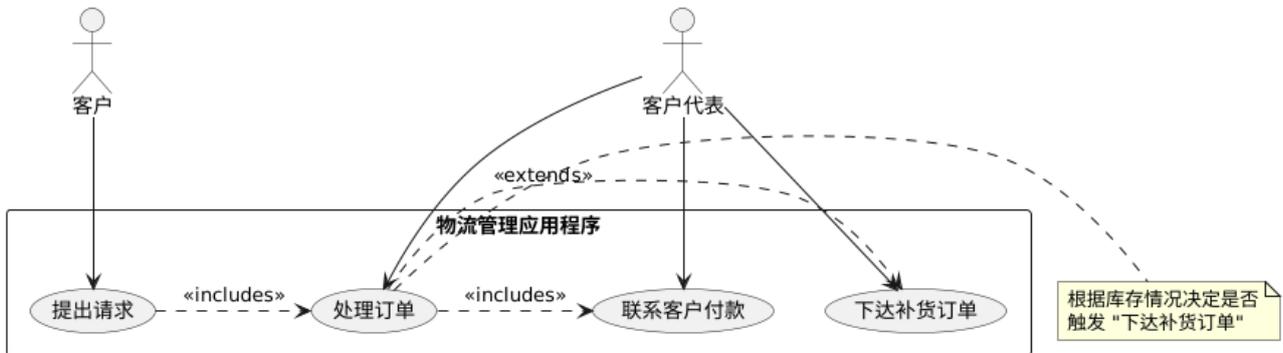


图 1: 用例图: 物流系统

3. 这些描述有助于避免哪些错误?

- \* 减少需求歧义、遗漏功能或错误的用户交互设计。

## 验证与确认 (Verification and Validation, V&V) 复习笔记

- 资源:
  - 附录 I: 验证与确认笔记 (Verification and Validation Notes)
- 工作计划:
  - 重点复习以下内容:
    - \* **静态与动态验证与确认 (Static and Dynamic V&V):**
      - 静态: 通过审查代码和文档 (如代码审查、静态分析) 发现错误, 无需运行程序。
      - 动态: 通过运行程序 (如测试) 验证系统行为。
    - \* **程序测试与调试 (Program Testing and Debugging):**
      - 测试: 运行程序以发现错误。
      - 调试: 定位和修复测试中发现的错误。
    - \* **验证与确认计划 (V&V Planning): 软件测试计划 (Software Test Plan)。**
      - 包括测试目标、范围、资源、测试用例和时间表。
    - \* **审查与自动静态分析 (Inspections and Automatic Static Analysis):**
      - 审查: 人工检查代码或文档。
      - 自动静态分析: 使用工具 (如 SonarQube) 检测代码中的潜在问题。

## 软件测试复习笔记

- 资源:
  - 附录 J: 软件测试笔记 (Software Testing Notes)
- 工作计划:
  - 重点复习以下内容:

\* **组件测试 (Component Testing) vs 集成测试 (Integration Testing):**

- 组件测试：测试单个模块或组件的功能。
- 集成测试：测试多个模块协同工作时的行为。

\* **测试用例结构 (Structure of a Test Case):**

- 数据 (Data)：测试输入。
- 结果 (Result)：预期输出。
- 报告 (Reports)：测试执行结果和问题记录。

\* **黑盒测试 (Black Box Testing):**

- 测试者不关心代码内部结构，仅根据输入和预期输出测试系统功能。

## 测试计划与测试类型问题

• **资源:**

- 附录 K: 测试计划笔记 (Test Plan Notes)

• **工作计划:**

- 阅读测试计划笔记并回答:

1. **功能测试 (Functional Testing) vs 集成测试 (Integration Testing) 的区别。**

- \* 功能测试：验证系统是否满足特定功能需求（如“用户可以登录”）。
- \* 集成测试：验证多个模块或组件一起工作时是否正常。

2. 假设你领导一个 20 人开发团队，开发第三版博客托管的网络平台，需重组团队并分配质量保证 (Quality Assurance, QA) 角色：

- \* **角色：**质量保证经理、测试工程师、开发工程师、自动化测试开发人员。

\* **任务:**

- 质量保证经理：制定测试计划，监督质量流程。
- 测试工程师：设计和执行测试用例。
- 开发工程师：编写代码并进行单元测试。
- 自动化测试开发人员：开发自动化测试脚本。

\* **假设与解释:**

- 假设平台已成熟，需重点关注回归测试 (Regression Testing)。
- 将质量保证任务分配给专门的测试团队，同时要求开发人员进行单元测试，以确保质量责任分散但有针对性。