**第一章思考题：**

**1、获取原始需求的传统方法和现代方法各有哪些？**

①传统：问卷调查、访谈、文档分析、头脑风暴、观察法

②现代：用户故事、原型制作、敏捷需求获取、场景分析、使用案例

**2、需求管理都包含哪些工作？如何执行的？**

①需求收集：从利益相关者那里获取项目的需求，这可能包括业务人员、用户、产品经理等，了解他们的期望和目标。

②需求分析：对收集到的需求进行详细的分析，确定其范围、优先级、可行性，并将其转化为明确、完整、一致的文档，如需求规格说明书。

③需求文档编写：撰写规范的需求文档，包括需求描述、功能设计、性能指标、限制条件等，以便于后续的设计和开发团队理解和实施。

④需求评审：组织会议让各方审查需求文档，确保理解一致，减少误解和遗漏。

⑤变更管理：记录并处理需求的变更请求，评估影响，更新需求文档，并通知相关人员。

⑥跟踪监控：在整个项目过程中持续跟踪已接受的需求，检查它们是否已经满足，以及是否需要调整。

⑦验收测试：在项目结束时，通过测试验证需求是否已得到满足，并由用户或客户验收。

**3、需求文档包含哪些内容？**

引言：项目背景、目的和范围的概述。

需求概述：对主要功能和需求的简要描述。

功能需求：详细列出系统的功能特性和用户交互。

非功能需求：包括性能、安全性、可用性等要求。

用户角色：定义不同用户类型及其需求。

用例：描述用户与系统的交互情景。

约束条件：技术、法律或业务限制。

需求优先级：需求的重要性和实现顺序。

附录：相关文档、术语定义和参考资料。

**第二章思考题：**

**一、现代软件工程如何处理软件开发所具有的本质 困难？**

1. 敏捷开发

描述：通过迭代和增量的方式快速交付功能，灵活应对需求变化。

实施：定期召开短期迭代会议，调整优先级和工作内容，确保持续反馈和改进。

2. 需求管理

描述：对需求进行系统化的收集、分析和变更管理。

实施：使用需求管理工具和文档，确保需求透明，并及时响应变化。

3. 自动化测试

描述：通过自动化测试工具提高测试效率，降低人工测试的错误率。

实施：构建持续集成（CI）和持续交付（CD）流水线，确保代码变更能自动化测试和部署。

4. 设计模式和架构

描述：应用设计模式和架构原则以解决常见问题，提高代码复用性和可维护性。

实施：在项目初期定义良好的系统架构，选用适当的设计模式，以支持可扩展性。

5. 版本控制

描述：使用版本控制系统管理代码变更，确保团队协作高效。

实施：通过Git等工具管理代码分支，记录历史版本，方便回滚和协作。

6. 文档化和知识共享

描述：保持良好的文档记录和知识共享，减少信息孤岛。

实施：使用维基、共享文档等工具，确保团队成员可以随时获取信息。

7. 团队协作和沟通

描述：通过团队协作工具提升团队间的沟通效率。

实施：定期召开站立会议、回顾会议，使用项目管理工具（如Jira、Trello）进行任务跟踪。

8. 用户参与

描述：积极邀请用户参与需求收集和测试过程，确保开发的产品符合用户期望。

实施：组织用户访谈、原型测试，获取用户反馈，进行需求调整。

**二、需求工程包含哪些主要的活动？如何理解“描 述”是需求工程的核心？ **

**（1）**需求获取：通过访谈、问卷和观察等方式收集用户和利益相关者的需求。

需求分析：对收集到的需求进行整理和分析，以识别关键需求和优先级。

需求规格说明：详细描述需求，以文档形式记录，确保所有利益相关者的理解一致。

需求验证：检查需求是否符合用户期望，确保需求的准确性和完整性。

需求管理：跟踪需求的变化，维护需求文档，确保版本控制和变更管理。

（2）清晰的描述确保所有利益相关者对系统需求的共同理解，从而减少误解和开发风险。良好的描述能够有效传达需求的具体性、可测性和可实现性，是成功开发的基础。

**三、结合本节讲义，完整的描述什么是需求**

在软件需求工程中，需求是指软件系统预期的功能、特性、性能以及其他特性的集合，它是软件开发的第一步。

**附加题**

**使用用例图描述需求的过程是怎样的？**

识别参与者: 首先确定系统的利益相关者或用户角色，例如客户、管理员、普通用户等，他们将成为用例图中的实体节点。

识别用例: 探索每个参与者可能执行的任务，即用例，这是系统的主要功能或行为。用例通常以动词开头，并简洁地描述其目的。

绘制用例图: 用例作为矩形框表示，参与者作为椭圆形表示，两者之间通过线连接，代表参与者与用例之间的关系。有时还会有包含多个用例的组合用例或扩展用例。

细化用例: 对复杂的用例进行分解，添加细节，比如前置条件、后置条件、基本流程和选择分支等

关联和消息传递: 显示参与者如何触发用例，以及用例内部的控制流。这可能涉及到消息的交换，也可能是事件驱动的。

验证和完善: 审查并确保用例图清晰、准确地表达出需求，必要时与其他团队成员讨论和确认。