1. 获取原始需求的传统方法和现代方法有那些？

传统方法：

面谈：

直接与利益相关者和用户交流，深入了解他们的需求和期望。

调查表：

通过结构化问卷收集数据，适用于大量用户反馈的收集。

观察：

观察用户在使用系统或产品时的实际行为，以发现潜在需求和问题。

研究业务文档：

分析已有的文档（如项目计划、用户手册）来提取需求信息。

现代方法：

原型法：

通过创建低保真或高保真的原型，允许用户提前体验产品的功能和界面。用户可以在原型上进行测试，提供反馈，帮助团队理解需求并快速迭代设计。

头脑风暴：

一种团队协作的创造性方法，成员自由地提出各种想法和解决方案。在没有批评的情况下，汇集多样的观点，从中筛选出潜在的需求和创新。

联合应用开发（JAD）：

一种结构化的需求获取方法，涉及利益相关者、用户和开发团队的共同参与。通过一系列会议，快速识别和定义系统需求，确保各方意见的充分表达与理解。

快速应用开发（RAD）：

强调快速开发和用户反馈的迭代过程。利用原型、用户反馈和并行开发来加速应用程序的开发，确保最终产品更符合用户需求。

1. 需求管理都包含哪些工作？如何执行的？

1. 标识、分类、组织需求并建立文档

执行方式：

需求识别：通过访谈、问卷和工作坊收集需求。

分类：将需求按功能、优先级和类型（如功能性与非功能性）进行分类，便于管理。

文档编写：使用标准模板编写需求文档，包括需求描述、来源、优先级及接受标准，确保信息清晰、易于理解。

2. 需求变更

执行方式：

变更请求：建立变更请求流程，任何需求变更都需通过正式的请求文档提出。

协商与确认：通过召开变更评审会议，相关利益方讨论变更的必要性和影响，达成共识后进行确认。

记录变更：更新需求文档和变更日志，详细记录变更内容、原因和批准情况，以确保透明和可追溯性。

3. 需求跟踪

执行方式：

需求追踪矩阵：创建需求追踪矩阵，将需求与设计、开发、测试用例等其他文档建立关系，便于跟踪需求的实现情况。

依赖关系管理：定期审查需求之间的依赖关系，确保任何变更都及时更新在相关文档中，保持一致性。

工具支持：使用需求管理工具（如JIRA、Trello）来管理和跟踪需求的状态，提供可视化支持，确保团队成员对需求状态有清晰了解。

1. 需求文档包含那些内容？

1. 引言

文档目的：说明文档的目标和范围。

背景信息：提供项目的背景和上下文。

2. 需求概述

业务需求：描述业务目标和项目的商业价值。

用户需求：概述目标用户及其需求。

3. 需求详细描述

功能需求：具体描述系统需实现的功能，包括输入、处理和输出。

非功能需求：如性能、可用性、安全性、可维护性等要求。

4. 需求优先级

优先级分类：将需求按重要性和紧急性分类（如高、中、低）。

5. 假设和约束

假设：列出在需求开发过程中所做的假设。

约束：描述项目中需遵守的限制条件（如技术、法规等）。

6. 接受标准

验收条件：定义每个需求的验收标准，确保需求实现后的验证依据。

7. 依赖关系

关联需求：指明需求之间的依赖关系及相关文档（如设计、测试用例）。

8. 变更管理

变更流程：描述需求变更的提出、评审和记录流程。

9. 附录

术语表：定义文档中使用的专业术语。

参考文献：列出参考的文档和资源。

10. 版本控制

版本历史：记录文档的版本号、日期及主要修改内容，确保追溯性。