**《软件工程导论》第三章作业**

1. **为什么迭代过程更容易管理变更?是不是本章所讨论的每一个敏捷过程都是迭代的？只用一次迭代就能完成项目的敏捷过程是否存在？说明原因。**

**答：**迭代过程之所以更容易管理变更，我认为有以下原因：

1. 短周期反馈：迭代的周期短，这样每个周期结束时都会有一个可工作的产品版本。这让团队和客户有机会在早期看到产品的进展和方向，从而及时提出更改或调整。
2. 减少浪费：对于变更要求，早期发现和处理总比项目结束时或后期发现要更为经济和高效。
3. 灵活的计划调整：在迭代结束后，团队会进行回顾和计划下一个迭代，这提供了调整方向和处理变更的机会。
4. 降低风险：因为在每次迭代结束时都会有一个可以工作的产品版本，所以即使在项目中途出现大的变更或重新定位，也不会导致之前的所有工作都被浪费。

虽然大部分敏捷过程都是迭代的，但也存在一些敏捷过程中的流程是增量的。增量和迭代是有区别的，迭代是指对一个功能或产品进行多次修订或优化，而增量则是指逐渐增加新的功能或组件到产品中。敏捷过程中，如Scrum和Extreme Programming (XP)就是迭代和增量的结合。

对于非常小的项目或明确、不变的需求，一次迭代可能是合理的。但在大多数情况下，敏捷过程需要多次迭代来充分发挥其优势。这样的情况更像是传统的瀑布模型，其中整个项目从开始到结束只进行一次阶段性的执行，而没有持续的反馈和调整。如果只用一次迭代来完成一个项目，那么这违背了敏捷的核心思想。

1. **写一个用户故事，描述“发货”这个用户故事。**

**答：**

作为一个在线商家，当我收到用户的购物订单后，我希望能够：

1. 在后台系统中快速地查找到该订单的所有详细信息，包括买家的地址、联系方式、订单中的商品以及商品数量。
2. 系统自动为我生成一个打印清单，列出需要从仓库中拿出的商品及其数量。
3. 在选取商品后，可以在系统中确认商品已经打包，并选择合适的物流公司为买家发货。
4. 系统自动为我生成一个包含物流追踪号的发货通知，这样我可以发送给买家，让他们知道商品已经在路上。
5. 如果遇到任何问题，比如说商品缺货或者物流公司出现故障，我希望系统可以及时提醒我，并给我提供备选方案。
6. 在商品成功送达买家后，能够收到系统的通知，这样我可以确保交易的顺利完成。

通过这样的发货流程，我可以确保商品能够在预期的时间内准确地到达用户手中，并让用户获得良好的购物体验。

验收标准：

1. 商家能够在系统中看到用户的订单详情，包括商品信息、数量、收货地址、预期到达时间等。
2. 商家可以选择合适的物流公司进行发货，并在系统中输入相关的物流信息。
3. 商家能在系统中更新订单状态为“已发货”，并为用户提供物流追踪信息。
4. 用户应能够在他们的账户中查看到订单的发货状态及物流追踪信息。
5. 商品应在预期的时间内送达用户手中。
6. **Scrum计划**

**1. 项目启动 (10月初)**

产品愿景: 与项目相关的所有利益相关者一起明确电商平台的目标和愿景。

产品待办事项清单: 创建一个初步的需求清单，描述电商平台的所有功能和需求。

**2. 组织Scrum团队 (10月第一周)**

指定产品负责人、Scrum Master以及确定开发团队成员。

产品负责人(Product Owner, PO): 负责定义产品的需求，优先排序待办事项，并确保团队了解需求。我们的Product Owner是任课老师。由任课老师组织讨论课来对各个模块的需求进行分析，提出新增和变更的需求并对工作量进行估算，划分出商品部分，支付模块，订单和其他部分。

Scrum Master: 确保Scrum流程得到遵循，协助团队解决障碍，并促进团队之间的沟通。保证各个不同角色之间的良好协作，消除障碍。帮助PO更好地利用团队的能力，指导并带领团队变得更加高效，实现更高质量。

开发团队: 由跨职能的成员组成，负责开发和交付产品。团队成员全职参与开发自我管理，没有头衔之分，不组建子团队。

1. **Sprint规划 (每月初)**

在Sprint规划会议上，由Product Owner 负责确定Product Backlog（按优先顺序排列的一个产品需求列表，排序越高约清晰具体，排序越低细节越少，是产品需求变动的唯一来源，动态，持续更新）；团队从Product Backlog中选择优先级高的任务，并估算在这个Sprint内可以完成的工作量。

执行细节：每次任务前团队先从Product Backlog列表中挑选出一个Story作为迭代目标，然后进行细化。然后估算当前sprint内要完成的工作并决定如何将选定的产品Backlog转化为潜在可交付产品的功能增量，即形成一个Sprint Backlog。保证无论是否发布，产品应该是可用的。

10月：

Sprint 1规划: 10月第一周

选择10月内要完成的任务，基于产品待办事项清单和团队的工作能力。

11月：

Sprint 2规划: 11月第一周

根据前一个Sprint的成果和反馈，选择11月要完成的任务。

12月：

Sprint 3规划: 12月第一周

选择最后一个Sprint的任务，确保项目结束前最重要的功能都被实现。

**4. 每日站立会议**

每天固定时间进行15分钟的站立会议，分享前一天的成果，计划当天的任务，并提及任何阻碍。Scrum Master负责解决任何阻碍，确保团队可以顺利工作。

**5. 工作进行中**

开发团队开始实施Sprint中的任务，按照Scrum流程保持透明和沟通。

产品负责人保持与开发团队的沟通，提供必要的需求解释和澄清。

做到每日集成，每天形成一个可以成功编译、并且可以演示的版本。

**6. Sprint检查 (每月末)**

在Sprint结束时，团队展示他们完成的工作给产品负责人和其他利益相关者。收集反馈并更新产品待办事项清单。

10月：

Sprint 1检查: 10月最后一天或11月第一天

11月：

Sprint 2检查: 11月最后一天或12月第一天

12月：

Sprint 3检查: 12月中旬或之前

确保有足够的时间进行项目的最终修补和完善。

**7. Sprint回顾 (每月末)**

在每个Sprint检查后，团队应进行回顾会议，讨论本月的成功和失败，总结并讨论需改进的地方，放入下一轮Sprint的产品需求中。

**8. 持续的产品待办事项维护**

产品负责人不断地更新产品待办事项清单，确保它反映了最新的需求。

**9. 项目结束 (12月末)**

完成所有功能的开发和测试。

进行项目回顾，总结经验教训。

根据需求，进行项目交付或部署。

**注意事项：**

考虑到项目总时长只有3个月，团队之间的沟通和合作显得尤为重要。确保每周有至少一次与产品负责人的沟通会议，以便快速解决需求上的疑问或不明确的地方。

为了更好地应对变化，考虑在每两周的中旬进行一次小的检查和回顾，以便更快地收集反馈并调整方向。

质量不能妥协。确保有良好的测试实践和代码审查机制。

确保每个Sprint都有明确的目标和结果，使团队始终对目标有清晰的认识。

团队每位成员需遵守承诺、勇气、专注、开发和尊重的价值观。除此之外，应当经常交付可工作的软件，间隔时间不可过长。