# 前言

知识整理是我最擅长的领域了。我一直认为每个人的学习方法是与众不同的，其在笔记上的对应体现也是不同的。对于我来说，我喜欢*倒着看书*。因为我有了不懂的概念，就等同于有了一个目标。通过自己的摸索解决目标的过程很有成就感，也不显得无聊。此外，作为一个学生，理解一个知识点往往是不够的。对于记忆，我想每个人也有自己的能力瓶颈。对于我来说，我一次只能记住少量的东西。所以我必须将知识拆分成*小块状*，再通过小块重构起一个知识体系。

以上两个原则贯穿了这份笔记的始终，共勉。

# 什么是CMMI？

递归图

图表, 气泡图

描述已自动生成

我印象很深刻的一次失败是一次编程作业。老师让我们写一个函数找到给定字符串的最长子串。我因为对函数调用还不是很熟悉，我硬生生利用指针实现了很多库函数的功能。学校好像知道我的难处一样，紧接着就开设了一门名为软件工程的课程。从中我了解到，编程不仅仅是代码，或者说，*代码只是编程的很小一个部分*。更加优秀的产品往往意味着更加复杂的功能。软件工程就是一个创造更加优秀产品的工具箱。

为了更好地理解软件工程，我决定从*程序的程序*这个角度来阐释。我们都知道程序由一系列步骤组成。软件工程就是一个输入为时间/精力，输出为软件的软件。这个“软件”的框架——步骤的整体性描述——被称为软件生命周期，或者软件过程。正如其他软件一样，*软件是一个需求的集合*。不是每个用户都需要它的每个功能。在应用软件工程时，也需要适当裁剪软件过程以适配时间/精力。

每个软件开发团队都是由人构成的，而每个人都是特殊的。所以，每个团队的软件过程也是不一样。随着团队经手的项目越来越多，他们越来越能找到适合自己的软件过程。在甲方寻找团队的过程中，一个重要的指标就是软件过程成熟度。既然每个团队的软件过程都是特殊的，甲方也很容易陷入“公说公有理，婆说婆有理”的困境。在这种历史背景下，CMMI应运而生。

CMMI全称为Capability Maturity Model Integration，由CMU的软件工程实验所开发，旨在评估一个团队的软件工程能力。

应老师的要求，我将评估我的软件能力与相应的改进措施。

## 改进措施模板

要提升每个过程域的能力，可以通过以下通用步骤：

标准化：制定和实施标准过程和实践。

文档化：记录过程和实践，确保一致性和可重复性。

培训：提供必要的培训，确保人员具备执行过程的能力。

监控和改进：定期评估过程性能，识别和实施改进措施。

## 需求管理

需求管理在很多方面和打猎很相似。

首先，在猎手出门时，猎手并不知道自己会打到什么猎物。其次，猎物是可以积累的。这对应了需求的收集阶段。猎物放久了可能会变质或者本身就不适合人类食用。同理，需求也需要变更。为了营养均衡猎手往往需要猎杀很多不同的猎物。同理，需求也需要进行分析和确认。

更加贴切的比喻是：需求工程师是摄影师，而需求是云朵。需求是变化的，需求工程师需要确定某一时刻的需求是什么样的。这更多的是商业层面的考虑。因为需求的不确定性，如果乙方承诺用户始终保持需求变更，这样的工作是不堪重负的。所以，商业项目不存在所谓的完美的软件产品，它们只是特定时期需求的产物。

想要改进这一方面的能力，我们只需要把措施模板映射到需求管理的定义上。

## 所有过程域及其改进

1. 需求管理 (REQM)

定义：确保项目的需求被正确管理和实现。

提升建议：

制定需求管理流程文档，确保所有需求被记录、分析和跟踪。

实施变更控制流程，确保需求变更得到适当处理和批准。

2. 项目计划 (PP)

定义：制定和维护项目计划，涵盖预算、时间表和资源分配。

提升建议：

使用项目管理软件来创建详细的项目计划，包含时间线、里程碑和资源分配。

定期审查和更新项目计划，以适应项目进展和变化。

3. 项目监控与控制 (PMC)

定义：监控和控制项目的执行，确保项目按计划进行。

提升建议：

建立项目监控系统，定期跟踪项目进度和预算。

定期召开项目状态会议，识别并解决问题。

4. 供应商协议管理 (SAM)

定义：管理与供应商的合同和关系，确保供应商交付符合要求。

提升建议：

制定供应商管理流程，包括评估、选择和监控供应商。

建立明确的合同和服务级别协议（SLA），并定期评估供应商绩效。

5. 测量与分析 (MA)

定义：开发和维护度量系统，用于支持管理和技术决策。

提升建议：

确定关键绩效指标（KPIs），并建立度量和分析系统。

定期收集和分析数据，用于改进过程和决策。

6. 过程与产品质量保证 (PPQA)

定义：确保过程和产品符合既定标准和要求。

提升建议：

制定质量保证计划，包含审计和评审流程。

定期进行过程审计和产品检查，确保符合质量标准。

7. 配置管理 (CM)

定义：管理项目的配置项，控制变更。

提升建议：

实施配置管理工具，记录和跟踪所有配置项和变更。

制定配置管理计划，包含变更控制流程。

8. 需求开发 (RD)

定义：开发和分析产品或项目的需求。

提升建议：

使用需求管理工具和技术，如用户故事、用例和原型。

与利益相关者密切合作，确保需求准确和完整。

9. 技术解决方案 (TS)

定义：设计、开发和实现技术解决方案。

提升建议：

制定详细的设计文档和技术规格。

使用原型和迭代开发方法，确保技术解决方案满足需求。

10. 产品集成 (PI)

定义：将组件集成为完整的产品。

提升建议：

建立集成测试环境，定期进行集成测试。

制定集成计划，包含集成步骤和时间表。

11. 验证 (VER)

定义：确认产品是否符合需求和规格。

提升建议：

制定详细的验证计划，包含测试用例和标准。

使用自动化测试工具，确保验证过程的效率和准确性。

12. 确认 (VAL)

定义：确认产品在实际使用环境中满足用户需求。

提升建议：

制定确认计划，包含实际使用场景的测试。

与用户紧密合作，确保产品在实际环境中的性能和适用性。

13. 组织过程焦点 (OPF)

定义：建立和改进组织的过程资产库。

提升建议：

创建过程改进小组，定期评估和改进过程。

制定和维护过程资产库，供所有项目参考和使用。

14. 组织过程定义 (OPD)

定义：定义和维护组织的标准过程。

提升建议：

制定标准过程文档，包含所有关键过程和最佳实践。

定期评估和更新标准过程，以适应组织的变化和需求。

15. 组织培训 (OT)

定义：提供必要的培训，确保人员具备执行过程的能力。

提升建议：

建立培训计划，包含必需的技能和知识。

定期评估培训效果，确保培训满足需求。

16. 集成项目管理 (IPM)

定义：管理项目和相关利益相关者，确保项目目标的实现。

提升建议：

制定项目管理计划，包含风险管理和沟通策略。

定期召开项目评审会议，确保所有利益相关者的期望和目标得到满足。

17. 风险管理 (RSKM)

定义：识别、分析和管理项目风险。

提升建议：

建立风险管理计划，包含风险识别、评估和应对策略。

定期进行风险评估和审查，确保风险得到有效管理。

18. 集成团队 (IT)

定义：建立跨职能团队，确保有效协作。

提升建议：

制定团队合作计划，包含沟通和协作策略。

定期进行团队建设活动，增强团队凝聚力和合作能力。

19. 量化项目管理 (QPM)

定义：使用统计和量化方法管理项目。

提升建议：

建立量化管理系统，收集和分析关键数据。

使用统计过程控制方法，监控和改进项目过程。

20. 组织过程性能 (OPP)

定义：建立和维护过程性能模型，支持项目和组织目标。

提升建议：

制定过程性能基准，评估和改进过程性能。

定期进行过程性能分析，识别和实施改进措施。

21. 因果分析与解决 (CAR)

定义：分析问题根本原因，实施有效解决方案。

提升建议：

建立因果分析流程，包含问题识别、分析和解决。

使用问题跟踪工具，确保所有问题得到及时和有效的解决。

22. 组织绩效管理 (OPM)

定义：通过过程改进提升组织绩效。

提升建议：

制定组织绩效改进计划，包含目标和措施。

定期评估组织绩效，识别和实施改进机会。