

CMMI 最早是应用于软件业的一个过程改进模型，为软件组织描述了从混乱的、不成熟的软件过程向成熟有序的软件过程进行改进的一条途径。CMMI 将软件过程的成熟度分为五个等级：1.初始级：软件过程基本无序，成功最主要依赖于个人努力。

2.已管理级：建立了基本的项目管理过程，能重复早先类似项目的成功经验。

3.已定义级：软件管理和工程过程文档化、标准化，并综合成组织的标准软件过程。

4.量化管理级：可以对软件过程和产品进行定量的理解与控制。

5.优化管理级：可以通过量化反馈和新技术持续改进过程。

在大一暑假实训时，曾开发一个“贪吃蛇”游戏软件，此软件根据 CMMI 层次成熟度模型的评估是属于初始级，开发编程过程中基本上没有系统化的管理以及管理文档的存在，最后的成果十分的“粗糙”，勉强符合基本需求。

为了使往后的开发更加系统化、标准化、规范化，根据“贪吃蛇”软件的开发提出改进建议。首先，在开发开始前，确立一个合适的项目管理工具是至关重要的。可以使用 GitHub 来管理项目，GitHub 可以帮助开发团队跟踪代码版本，确保每一次修改和更新都有清晰的记录。对于日后进行版本回溯、问题追溯，以及团队协作至关重要，还可以避免文件丢失或误操作带来的问题。此外，分支管理功能也能够帮助开发者分工协作，避免不同开发任务之间的冲突。在软件开发的初期阶段，常常因为需求不明确或频繁变化导致开发过程混乱，影响项目进度。为了避免这一情况的发生，应该在开发前期就建立一份《需求规格说明书》，对软件所需要实现的功能点进行详细记录，避免因口头讨论导致的需求遗漏或误解。对于需求变更，应当在说明书中做出详细的记录，保证每一个变更都能够被追踪和管理，从而保证开发的规范性和有效性。并且为防止盲目开发，应该做一个开发任务项目计划，如：可以使用甘特图规划开发周期，可视化进度，将任务的各个阶段、时间节点、负责人和目标进行清晰划分，便于汇报总结。此外，因为是多人合作开发一款软件，防止因格式不统一而造成的合并冲突，应该设立一个《代码规范文档》，规范代码变量的使用，以及对函数注释进行规范化的确立，这样可以增强代码可读性，减少调试时间；并且可以在每日工作前开一个交流会议，了解开发进度，确认成员是否遇到了技术问题。在编码时，要模块化代码结构，这样做便于分工，并且在修改时不会影响到其余代码，提高了代码复用性。软件开发中，bug 的出现是无法避免的，因此，进行有效的缺陷管理至关重要。在项目开发过程中，可以使用 Excel 表格或类似工具记录每一个缺陷的详细信息，包括缺陷出现的位置、原因和修改方法。通过这种方式，可以不断总结经验，避免同样的错误重复出现。同时，在软件测试阶段，要特别注意增加测试覆盖率统计，确保软件的核心功能能够被充分测试，及时发现潜在的盲区，避免在最终发布时出现严重问题。