



《软件过程与管理》期末论文



学 院:	软件学院
专 业:	软件工程
学生姓名:	郭 政
学 号:	2023141461076
指导老师:	毋攀良
成 绩:	

二零二五 年 六 月 八 日

基于 CMMI 模型的软件开发过程评估与改进计划

学院：软件学院

姓名：郭政

学号：2023141461076

一、CMMI 层次成熟度模型简述

CMMI 发源于美国卡内基梅隆大学的软件工程研究所，如今已经是全球广泛认可的一种评估和改进软件开发组织能力的重要工具。

该模型将组织的过程能力划分为五个逐步递进的成熟度等级：

初始级（Level 1）：这个阶段更像是在“摸着石头过河”，开发过程没什么固定章法，基本依赖于个人经验或临时的安排，可以说是相当随性，项目结果常常难以预测。

已管理级（Level 2）：在此阶段，组织已具备初步的项目管理能力，能够通过计划与跟踪手段控制进度与资源。

已定义级（Level 3）：此时不再满足于项目级的计划管控。组织会花力气建立起一套标准化的、适用于所有项目的开发流程规范，并形成正式的文档。

量化管理级（Level 4）：在此阶段，组织对开发过程与产品质量实施量化控制，收集并分析相关过程数据，用数据支持决策与风险预警。

优化级（Level 5）：最高成熟度等级，代表组织已建立持续改进机制，能主动识别过程瓶颈并运用创新技术或管理手段实现优化。

CMMI 的核心价值在于：它像一把清晰的标尺，让组织能客观识别自己当前处于哪个管理成熟度水平。看清了位置，就能更有针对性地补齐短板，优化开发流程，最终目标是从“勉强完成任务”蜕变为“持续稳定地交付高质量产品”。

二、个人开发项目回顾与成熟度评估

学习期间我参加了多项软件开发实践，其中一个比较有代表性的是基于 OCR 技术的网站平台开发。该平台致力于实现对图像文档的精准文字提取、公式识别、表格重建，最终支持用户在线编辑文档内容并导出结构化结果。

下面从 CMMI 模型角度评估，我认为整体开发过程主要处于 Level2 水平，具备初步的过程可控性，但仍有明显规范化欠缺：

1. 项目计划与开发进度控制：项目初期我们借助 Trello 工具进行任务划分与进度跟踪，尽管具备基本的计划意识，但缺乏正式的时间里程碑与风险缓解方案，易在集成阶段暴露依赖性问题。

2. 需求管理过程：用户需求最初来源于课程设想与小组讨论，主要依赖非正式会议记录与群聊协商，未形成系统的需求文档。这导致后期需求变更时缺

乏可追踪依据，功能变动容易引发团队理解偏差与实现重复。

3. 软件过程监控与度量：项目中虽设有定期会议同步开发状态，部分关键模块在提交前会进行同伴代码审查，但整体缺乏量化指标支持。

4. 版本与配置管理：团队统一采用 Git 进行代码版本控制，制定了基本的分支管理策略，但是存在重要模型文件与配置项未纳入系统性配置管理范畴的问题。



图 1 前端截图

5. 测试与质量保证：由于平台功能涵盖识别引擎、接口服务、数据处理与前端渲染多个层次，理论上应构建完整的多层测试框架。但实际仅进行部分手动测试与接口验证，缺乏自动化测试用例、回归测试脚本或压力测试机制。

我认为，我的项目体现出团队良好的主动协作意识与工具使用习惯，能够对任务进行基本分配与监控，具备一定可控性和重复性特征。然而，在标准流

程建立、数据驱动优化、文档管理等方面仍缺乏组织级制度化支撑，因而整体仍处于 CMMI 的 Level2 阶段。

三、改进计划

虽然我们团队能靠工具实现基本的管理，卡在 CMMI Level 2 上其实反映出核心问题：太依赖个人经验和临时安排，缺乏组织级的统一标准和制度。要真正迈进 Level 3，光靠自觉不够，得建立起一套团队公认的可复用流程。基于项目反思，我提议几个关键的改进方向：

1. 先把规则定下来：制度化流程，统一认知

先要制定一套清晰完整的项目流程，每个阶段配套标准模板，项目启动时就得进行培训，让所有人明确知道每一步该干嘛、该交什么、标准是什么。目标是大家做事的基本套路统一起来。

2. 迭代中见真章：引入敏捷机制增强响应

项目周期长容易失控。借鉴 Scrum 或 Kanban 这类敏捷方法是个好选择。把项目拆成一个个短周期迭代，每个迭代聚焦几个具体的用户故事目标。这种小步快跑的方式能及时暴露问题、快速调整，项目可控性会强很多。

3. 用数据说话：建立量化指标支撑决策

管理光靠感觉不牢靠。得开始量化和收集数据：代码覆盖率、缺陷密度、提交频率、迭代周期多长、单元测试通过率、构建失败频率等等。靠工具自动统计和报告这些指标，让团队能真正看清问题在哪，预测风险，做决策也有依据。

4. 安全网要织牢：构建多层级自动化测试体系

手动测试明显不够用，特别是系统越来越复杂。必须下决心建立覆盖单元测试、接口测试、系统测试和回归测试的自动化测试体系。目标是把这些测试嵌入持续集成/持续部署流程里，形成每次代码提交都自动验证的机制。

5. 经验不能丢：沉淀知识，从个人到组织财富

建立一个共享的知识库，把标准流程文档、常见问题解决方案、技术规范指南、项目复盘记录都系统化整理、留存下来。定期做内部分享，让“个人经验”真正沉淀为“团队资产”。

通过这一系列措施，建立起团队可共享、可复用、基于数据和工具驱动的标准化流程，让我们的项目管理能力从“管得住”（Level 2）真正跨到“有章法”（Level 3）的台阶上。

四、结语

对软件开发过程进行评估，其意义不仅在于判断当前所处的成熟度阶段，更在于明确未来可以改进的方向与路径。在本次分析中，我结合项目实际情

况，提出了包括流程制度化、引入敏捷机制、完善需求管理、建立量化指标、强化测试体系以及积累知识资产在内的六项改进建议。这些措施正好对应了当前开发过程中存在的薄弱环节，是我们迈向更高过程成熟度的重要起点。

提升开发成熟度并非一蹴而就的过程，它需要时间积累、持续反馈和团队共同努力。未来的项目实践中，我将持续关注过程优化与工程化管理，并将此次反思转化为行动计划，推动自己和团队向着更专业、更稳定的软件开发目标不断迈进。

道阻且长，行则将至。只要方向明确、步伐坚定，前方的成长就不会遥远。