

四川大學

SICHUAN UNIVERSITY



题 目 基于 CMMI 模型的软件过程成熟度评估与
改进研究

学 院 软件学院

学生姓名 张焱

专 业 软件工程

学 号 2022141461065 年 级 2022

指导教师 毋攀良

2025 年 6 月 5 日

摘要：本文基于 CMMI 成熟度模型，对 Spring Boot 学生管理系统开发项目中的软件过程成熟度进行评估，并提出相应的过程改进计划。通过分析 CMMI 五个成熟度级别，结合实际开发经验，识别当前开发过程的不足，制定系统性改进方案。

1. CMMI 层次成熟度模型概述

CMMI (Capability Maturity Model Integration) 是软件过程改进模型，将软件过程成熟度划分为五个递进层次。

1.1 初始级 (Level 1)

过程缺乏标准化和可预测性，开发工作依赖个人能力，过程临时混乱，项目成功难以预测。

1.2 管理级 (Level 2)

建立基本项目管理过程，包括需求管理、项目计划、监控控制、配置管理等，过程具备基本可重复性。

1.3 已定义级 (Level 3)

建立标准化过程并在组织范围内一致应用，包括需求开发、技术解决方案、产品集成、验证确认等过程域。

1.4 量化管理级 (Level 4)

使用统计和定量技术控制过程性能，通过数据分析预测项目结果，实现定量过程管理。

1.5 优化级 (Level 5)

持续过程改进和创新，通过创新技术和过程变更实现量化改进目标，具有强烈改进文化。

2. Spring Boot 学生管理系统开发过程成熟度评估

2.1 项目背景

该学生管理系统基于 Spring Boot 框架开发，主要功能包括学生信息管理、课程管理、成绩管理和用户权限管理。使用 MySQL 数据库，前端采用 Thymeleaf 和 Bootstrap。开发周期 8 周，涉及需求分析、系统设计、编码实现、测试和部署等阶段。

2.2 当前成熟度评估

通过开发过程回顾分析，当前软件过程成熟度介于 CMMI Level 1 和 Level 2 之间，倾向于 Level 1.5 状态。

Level 1 特征体现：需求管理方面，缺乏正式需求文档和变更控制机制，需求变化通过口头沟通记录，导致后期功能理解偏差需要返工。项目计划相对粗糙，缺乏详细工作分解结构和里程碑，时间估算基于直觉，部分功能开发超出预期。质量管理缺乏系统化测试计划，测试临时性强，无完整测试文档和缺陷跟踪。

Level 2 特征萌芽：配置管理使用 Git 版本控制，建立基本分支管理策略。项目监控方面定期检查开发进度，具备基本跟踪意识。文档管理开始重视技术文档编写，包括数据库设计和 API 接口文档。风险管理能识别技术风险并采取应对措施。

2.3 成熟度评估总结

当前软件过程成熟度主要表现为：过程可重复性不足，缺乏标准化开发流程；文档化程度较低，不够系统完整；质量控制薄弱，缺乏正式质量保证过程；项目管理基础薄弱，计划和跟踪不够正式系统。因此判断当前成熟度水平为 Level 1.5，正在从初始级向管理级过渡。

3. 过程改进计划

基于当前成熟度评估结果，制定系统性过程改进计划，目标是达到 CMMI Level 2 成熟度水平。

3.1 短期改进目标（1-3 个月）

建立基础项目管理过程 建立标准化项目启动流程，制定项目计划模板，包含工作分解结构、时间估算、资源分配和风险识别。建立项目监控机制，包括周报制度、里程碑检查点和进度跟踪工具。

完善需求管理过程 建立需求收集分析标准流程，包括需求调研方法、需求文档模板和需求评审机制。实施需求变更控制流程，建立需求基线管理和影响分析机制，建立需求追溯矩阵确保全程可追溯性。

强化配置管理实践 完善 Git 分支管理策略，制定代码提交规范，包括提交信息格式、代码审查要求和合并策略。建立构建和部署自动化流程，减少人工操作错误。

3.2 中期改进目标（3-6 个月）

建立质量保证体系 制定代码审查标准和流程，建立单元测试、集成测试和系统测试标准流程，制定测试用例设计规范。实施缺陷管理流程，建立缺陷分类、优先级定义和处理流程。

实施度量与分析 建立基本项目度量体系，包括进度指标、质量指标和生产率指标。建立数据收集和分析流程，定期生成项目报告和趋势分析，建立度量数据库为未来项目提供历史数据支持。

完善文档管理 建立项目文档管理体系，制定各类文档模板，建立文档审查和更新机制，确保文档准确性和时效性。

3.3 长期改进目标（6-12 个月）

过程标准化 制定组织级软件开发过程标准，建立过程资产库，收集整理可重用的过程模板、检查表和指导文档。实施过程培训计划，确保团队成员掌握标准过程。

持续改进机制 建立过程改进建议收集和处理机制，定期进行过程评估回顾，识别问题和改进机会。建立改进效果跟踪机制，评估改进措施有效性。

3.4 改进实施策略

采用渐进式改进策略，优先实施影响大、实施难度小的改进措施。引入适当工具支持过程改进，建立工具使用培训机制。通过培训宣传建立质量意识和过程改进文化，营造持续改进氛围。

4. 结论

通过对 CMMI 成熟度模型的分析和 Spring Boot 学生管理系统开发过程的评估，明确了当前软件过程的成熟度水平和主要问题。制定的改进计划涵盖项目管理、需求管理、配置管理、质量保证和度量分析等关键过程域，为提升软件开发过程成熟度提供了系统性解决方案。

过程改进是长期系统工程，需要持续投入和坚持。通过逐步实施改进计划，不仅能够提

升软件开发质量和效率，还能积累宝贵的过程管理经验。在实施过程中应保持灵活性，根据实际效果不断优化改进措施，确保过程改进真正提升项目绩效。

本次研究深刻认识到软件过程成熟度对项目成功的重要影响，为今后的软件开发实践提供了明确的改进方向和实施路径。