

《软件过程与管理》课程论文

一、CMMI 层次成熟度模型简述

成熟度等级模型（Staged Representation），共分五个层级，由低到高标志着过程管理能力与可预测性的提升：

1. 初始级（Level 1 - Initial）：

- **特征：** 过程是临时的、混乱的、反应式的。成功高度依赖个人能力和英雄主义行为。项目常超期、超预算，质量不稳定。几乎没有标准化流程。
- **状态：** “混沌无序”。

2. 管理级（Level 2 - Managed）：

- **特征：** 在项目层面建立了基本的管理过程。能对项目进行规划、监督和控制。需求、计划、承诺、配置、质量保证等核心项目管理过程域得到实践。目标是确保项目可控、可重复。
- **状态：** “项目可控”。

3. 定义级（Level 3 - Defined）：

- **特征：** 在组织层面建立了标准化的过程体系（Organization's Standard Process, OSP）。所有项目都根据组织标准裁剪并执行经过明确定义、文档化的过程。过程得到更深入的理解、管理和一致性，强调工程过程（如设计、测试）的规范化。
- **状态：** “组织规范”。

4. 量化管理级（Level 4 - Quantitatively Managed）：

- **特征：** 对过程和产品质量建立了**量化目标**。利用统计技术和过程性能模型，对关键子过程和产品质量进行量化管理、预测和控制。决策基于数据和客观分析，减少主观判断。
- **状态：** “数据驱动”。

5. 优化级（Level 5 - Optimizing）：

- **特征：** 聚焦于**持续的过程改进**。通过识别过程弱点和创新机会，运用量化反馈和试点新技术，主动、系统地优化过程性能。目标是提升效率、预防缺陷、适应变化。

- **状态：**“持续改进”。

每个成熟度等级都是下一等级的基础，组织需逐级提升，无法跨越。每个等级包含一组特定的**过程域（Process Areas, PAs）**，组织需满足这些过程域的所有**特定目标（Specific Goals, SGs）**和**共性目标（Generic Goals, GGs）**才能达到该等级。[1]

二、项目过程成熟度评估

本次软件开发过程成熟度评估选择《软件项目管理》课程我和项目小组所做的“川大集合吧”软件。基于项目需求文档、软件项目课程报告等项目产出文档分析，“川大集合吧”团队成熟度介于 **Level 2（可重复级）** 向 **Level 3（已定义级）** 过渡，具体表现如下：

成熟实践（符合 Level 2~3）

- **项目管理**
 - 明确阶段划分（需求→设计→开发→测试→运维），制定甘特图里程碑。
 - 风险管理全面：识别 9 类风险（需求变更、技术、安全等），制定应对策略（如敏捷响应需求变化、预留缓冲时间）。
- **需求工程**
 - 需求说明书结构完整（功能/非功能需求、接口定义），用户分类清晰。
 - 采用敏捷开发，通过迭代响应用户反馈。
- **技术规范性**
 - 技术栈统一（SSH+MySQL+阿里云+H5），安全措施严格（密码加密 SHA-256+Salt、防御 SQL 注入）。

薄弱环节（未达 Level 3）

1. **过程标准化不足**
 - 设计文档未细化模块接口，代码审查、编码规范未提及。
 - 测试仅 5 天，未覆盖全场景（如未实现室内导航却无替代方案测试）。
2. **量化管理缺失**
 - 性能目标（并发 ≥ 5000 ）未达标（实际测试仅 1000 并发），但未分析瓶颈原因。

- 需求变更无跟踪机制（如地图 API、积分商城供应商问题悬而未决）。

3. 组织级过程资产缺失

- 知识传递依赖文档，但无共享知识库或培训体系。
- 无配置管理（代码/文档版本控制未提及）。

结论：团队具备基础项目管理能力（Level 2），但过程未标准化、数据驱动不足，需向 Level 3 提升。

三、过程改进计划

目标：1 年内达成 Level 3（已定义级）

改进领域	具体措施	时间节点
1. 过程标准化	<ul style="list-style-type: none">制定《详细设计模板》强制模块接口说明（输入/输出/异常处理）建立《编码规范》并引入 SonarQube 自动审查	Q3 2025
2. 需求管理	<ul style="list-style-type: none">使用 Jira 管理需求跟踪矩阵（RTM），关联测试用例每月评审待定问题（如地图 API），明确责任人/解决路径	Q2 2025
3. 量化控制	<ul style="list-style-type: none">定义性能基线（响应时间/并发数）部署 Prometheus 监控生产环境，每周生成性能报告测试周期延长至 2 周，引入 JMeter 自动化压测	Q4 2025
4. 组织级资产	<ul style="list-style-type: none">搭建 GitLab 知识库（设计文档/代码片段/FAQ）季度技术培训（如阿里云优化、安全攻防演练）	Q1 2026
5. 配置管理	<ul style="list-style-type: none">强制 Git 分支策略（main/dev/feature），所有文档版本化存储	立即实施
预期收益		

- 质量提升：缺陷率降低 30%（通过标准化设计+代码审查）。

- **效率提升**：需求变更响应速度提高 50%（Jira 跟踪+敏捷迭代）。
- **风险可控**：性能瓶颈提前暴露（监控+自动化测试），避免上线故障。

总结

“川大集合吧”项目已具备可重复级（Level 2）的核心能力，尤其在风险管理和技术选型上表现突出。但需通过**过程标准化**、**数据驱动决策**和**组织资产沉淀**突破 Level 3 瓶颈。改进计划聚焦可落地的工具链（Jira/GitLab/SonarQube）和制度（设计规范/量化监控），最终实现过程可控、质量可测、知识可传承的工程化目标。

参考文献

[1] 如是说如是.软件能力成熟度模型（CMMI）.[EB/OL].(2019-04-30).[2025-06-03]. [软件能力成熟度模型（CMMI）_cmmi 模型-CSDN 博客](#)