

# 四川大學



题    目 \_\_\_\_\_ 学期论文 \_\_\_\_\_

学    院 \_\_\_\_\_ 软件学院 \_\_\_\_\_

专    业 \_\_\_\_\_ 软件工程 \_\_\_\_\_

学    生 \_\_\_\_\_ 黄煜程 \_\_\_\_\_

学    号 \_\_\_\_\_ **2023141461041** \_\_\_\_\_

年    级 \_\_\_\_\_ 大二 \_\_\_\_\_

课程名称 \_\_\_\_\_ 软件过程与管理 \_\_\_\_\_

任课教师 \_\_\_\_\_ 毋攀良 \_\_\_\_\_

**2025 年 6 月**

# 1. CMMI 成熟度模型简述

CMMI 是用于对组织过程的成熟度进行评估, 并提供过程改进指导原则的能力成熟度模型。此外, CMMI 作为风险管理模型, 提供了一种衡量组织风险管理能力的方法。

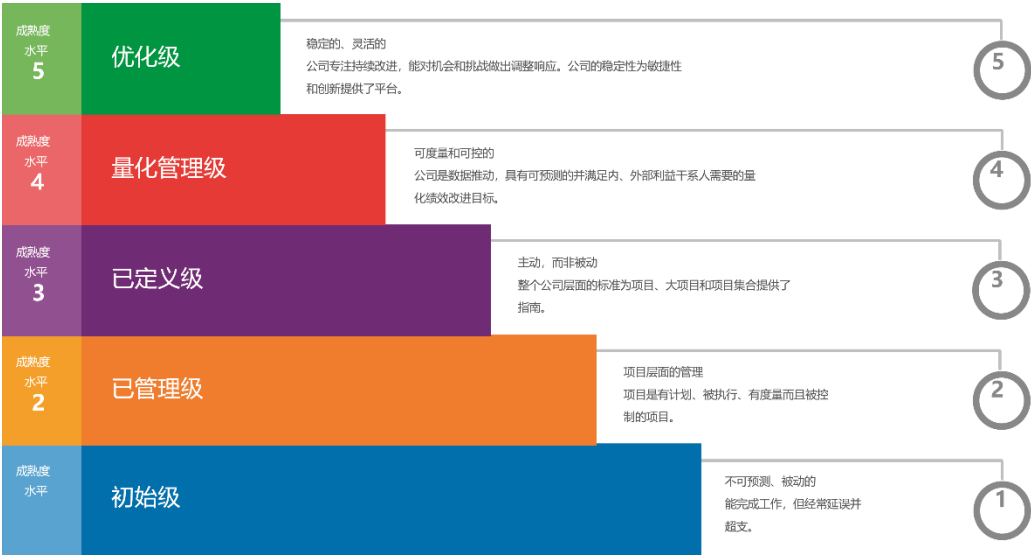


图 1 CMMI 成熟度模型

如上图所示, CMMI 共分为五个等级: 初始级, 已管理级, 已定义级, 量化管理级, 和优化级。

## 1.1. 初始级

初始级的过程无序, 混乱, 缺乏标准化流程。项目执行依赖个人能力, 而非制度化过程。其管理方式缺乏预见性和规划。因此, 初始级的成功率高度依赖团队核心人员, 难以把控项目进度与质量, 缺乏有效监控机制, 过程不可重复且难以预测。

## 1.2. 已管理级

已管理级在初始级的基础上建立了基础项目管理流程, 并通过标准化文档实现过程课重复。该阶段成本, 进度和质量可监控, 能够基于历史经验管理新项目, 降低一定风

险.

### 1.3. 已定义级

已定义级的过程全面标准化并文档化, 形成了组织级标准流程库. 所有项目需要基于组织标准过程进行裁剪执行, 同时建立培训体系, 确保员工掌握标准过程. 在此阶段, 开发过程, 成本, 计划和功能得到进一步监控与管理, 软件质量得到显著提升. 同时过程上升为组织资产.

### 1.4. 量化管理级

量化管理级引入量化管理方法, 对关键指标进行数据驱动分析. 量化管理级建立过程数据库, 实现过程和产品质量的定量控制, 并据此预测产品进度与性能趋势, 能够及时采取纠正措施.

### 1.5. 优化级

优化级是 CMMI 最高级别, 它在前四集的基础上持续优化过程, 采用新技术和方法提升效率和质量, 通过量化反馈识别改进机会, 主动预防缺陷和薄弱环节. 此阶段过程本身已成为企业战略核心, 追求持续绩效提升.

## 2. 项目成熟度评估

在数据库系统和信息管理课程设计中, 我通过 javascript 语言与 sqlite 数据库模块开发了图书管理系统. 具体而言, 基于原项目已完成框架, 我设计并构建了 13 个相关 API, 项目结构如下图所示.

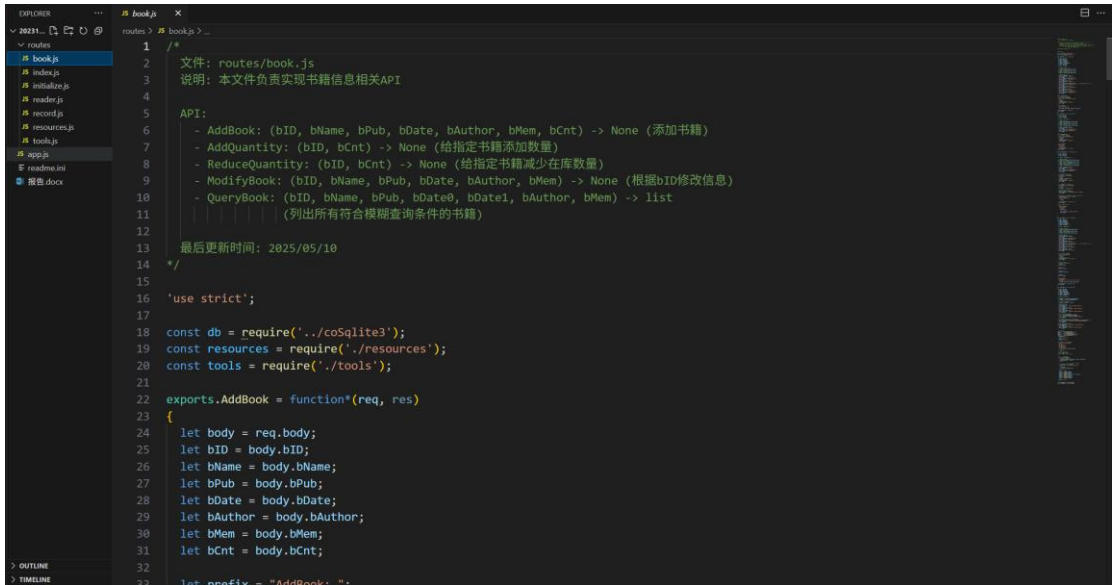


图 2 数据库课程设计

根据前述 **CMMI 成熟度模型简述**, 该项目仅达到**初始级**水平. 原因如下:

1. 项目缺乏明确计划, 未建立项目开发过程.
2. 项目缺乏监控与配置管理.
3. 项目无相应文档化过程.
4. 项目开发经验未进行记录, 不具有可重复性.
5. 项目缺乏自动化流程, 开发过程较为被动.

## 3. 项目改进计划

根据前述成熟度分析的五条原因, 我制定了以下改进计划, 共分为**现状评估与差距分析**, **建立标准化过程**, 采用**软件管理开发**三个部分.

### 3.1. 现状评估与差距分析

对照 CMMI 成熟度模型与 ISO 12207 标准, 将本项目涉及到的所有软件活动进行

定义与总结, 并评估当前过程环节缺失部分, 以识别出该项目在过程标准化, 文档管理, 质量确认等活动中的不足, 从而确定需要改进的部分. 譬如, 本项目缺乏配置管理, 对源代码进行修改时未使用 git 等版本控制工具, 导致版本变更与回溯繁琐且抵消.

### 3.2. 建立标准化过程

根据前述标准文档与现状评估结果, 对软件过程标准进行裁剪, 以建立一套适配本项目的标准化过程. 由于 API 内容与格式已由课程明确给出规范, 因此本项目在实际开发过程中主要完成设计, 编码与测试环节. 然而, 由于对软件过程认知的不足, 本项目在开发过程仅有编码环节得到重视, 而忽视了设计与测试环节的重要性, 从而导致开发难度随代码量增加而指数增长. 本计划中, 该项目应分为设计, 编码, 测试三个环节, 并辅以配置管理, 文档化贯穿整个软件生命周期, 采用测试驱动编码的策略提高编码的效率与质量.

### 3.3. 采用软件管理开发

如**项目成熟度评估**部分所述, 本项目缺乏软件管理开发的意识, 从而导致了软件开发的时间与人力成本显著提升. 经过调研, 本项目的软件管理开发部分宜采用 PingCode 进行软件开发管理与测试用例管理, 使用 git 进行版本控制, 并将项目托管在 github 上; 使用 plantuml 实现代码绘制 UML 图, 提前规划项目架构.

通过实施以上改进计划, 经初步估算本项目可达到 CMMI 二级, 即**已管理级**水平, 并为向**已定义级**进一步提升奠定管理基础.