# CMMI层次成熟度模型：

CMMI应该是能力成熟度模型集成（Capability Maturity Model Integration）的缩写，它主要用于帮助组织改进其流程，特别是在软件开发、产品和服务方面。CMMI有五个级别，从1到5，每个级别代表不同的成熟度。

## 初始级（Initial）

* 特点：流程无序且灵活多变，依赖个人能力与临时应对，项目结果不可预测。
* 关键问题：组织缺乏稳定性，高度依赖“英雄式”人物，风险较高。
* 改进方向：建立基本流程意识，减少混乱。

## 受管理级（Managed/Repeated）

* 特点：引入项目管理流程，确保项目目标达成。流程在类似项目中可重复应用。
* 关键问题：需求管理、项目计划、监控与跟踪（如成本、进度）。
* 改进方向：通过标准化项目管理提升可预测性。

## 定义级（Defined）

* 特点：流程在组织范围内标准化，并文档化为指南和模板。
* 关键问题：集成软件工程与管理流程（如设计、测试、质量保证），支持跨项目一致性。
* 改进方向：实现流程制度化，提升效率与质量。

## 量化管理级（Quantitatively Managed）

* 特点：基于量化数据进行流程管理，使用统计技术监控和调整过程性能。
* 关键问题：建立流程和产品质量的量化目标（如缺陷率、生产率），实施统计过程控制。
* 改进方向：通过数据驱动决策优化流程稳定性。

## 优化级（Optimizing）

* 特点：以持续创新为核心，通过渐进式改进和技术变革提升流程能力。
* 关键问题：识别流程缺陷并主动改进（如根因分析、技术试点），适应快速变化的业务需求。
* 改进方向：实现敏捷性与卓越绩效。

# 评估过往项目的成熟度

在软件项目管理课程上，我们完成了一个“Stream游戏推荐平台”。项目核心内容为开发一个多功能的游戏推荐系统，具体包括用户注册、登录等基础用户功能，以及首页轮播推荐、排行榜、AI 游戏推荐助手等游戏推荐功能。

技术实现上，前端基于 React + Tailwind CSS 实现响应式界面；后端使用 Next.js 构建 API，集成 Supabase 数据库存储用户与游戏数据；AI 模块，通过 Vercel AI SDK 连接 LLM，实现智能对话与游戏推荐。

按照CMMI的评估体系，这个项目应该处于第二、第三阶段。

**优点：**

文档规范：我们的项目具有项目计划书、需求分析、概要设计、测试文档、部署手册、使用手册等文档。

分工合理，小组成员都具有一定工作量，都按时交付，成员技能适合于分工模块，设计阶段的全部功能都成功实现。

每周上课时，组长会召开会议，分享各成员进度，并且安排后续分工。

技术标准化，采用Next.js、Tailwind CSS、Supabase等统一技术栈，提升开发效率。

**缺点：**

沟通机制仅通过文档传递，可以进一步完善。

进度监控颗粒度不足。

各功能之间的接口没有形成文档和对齐，执行流程需细化。

测试部分出现的问题仅提出优化方向，未形成闭环改进机制。

# 项目成熟度改进方案

* 性能测试仅关注响应时间，缺乏缺陷密度、代码覆盖率等全流程指标。
* 建立数据驱动的持续改进机制，实现流程与业务目标的动态适配。
* 引入代码质量指标：代码覆盖率（C80%↑）、静态代码分析（SonarQube）。2. 缺陷管理：记录缺陷类型、修复周期、根因分类（如接口不一致、AI模型延迟）。
* 制定接口规范：使用Protobuf或Swagger定义模块间数据格式。建立技术对齐机制：每周召开接口评审会议，同步依赖关系。
* 文档版本控制：采用Git管理用户手册、部署文档。动态更新：根据测试反馈实时修订文档。
* 细化任务分解：将甘特图拆解为子任务（如AI模块拆分为模型调用、缓存策略）。增加每日组会：同步模块进展，暴露风险点。
* 搭建团队Wiki：汇总技术踩坑记录、测试工具使用指南。定期分享会：复盘项目教训。