1. **CMMI层次成熟度模型简述**

能力成熟度模型集成(Capability Maturity Model Integration，CMMI)是由美国卡内基梅隆大学软件工程研究所(SEI)开发的过程改进模型，广泛应用于全球软件开发组织。该模型将组织的过程能力划分为五个渐进式成熟度等级，每个等级代表着一组过程域的达成程度。

CMMI能力成熟度模型集成分为5个级别：

1. **初始级(Level 1 - Initial)**：过程不可预测且难以控制，成功依赖个人能力，进度和预算经常超支，缺乏系统的需求管理和变更控制
2. **可重复级(Level 2 - Managed)**：建立了基本项目管理过程，能重复早期成功经验，进行基本的质量保证活动，项目计划包含明确的里程碑
3. **已定义级(Level 3 - Defined)**：过程已标准化、文档化，并能适应组织需求，开展系统的培训计划，实施同行评审和技术评审
4. **量化管理级(Level 4 - Quantitatively Managed)**：过程可测量和控制，基于统计数据进行管理，开展根本原因分析
5. **优化级(Level 5 - Optimizing)**：持续改进过程，能识别并消除过程缺陷根源，技术创新系统化，采用自动化过程优化，实施全组织的知识管理

**二、过往项目过程成熟度评估**

基于我参与的SpringBoot+Vue医院挂号管理系统开发经验，评估当前成熟度约为**Level 2(可重复级)**，具体表现：

**已具备的Level 2特征：**

1. 需求管理：有基本的需求文档和变更记录
2. 项目计划：制定了开发计划和里程碑
3. 项目监控：定期进行进度跟踪和会议
4. 配置管理：使用Git进行代码版本控制
5. 质量保证：进行了基本的测试(单元测试、功能测试)

**未达到的Level 3特征：**

1. 缺乏标准化的开发过程和文档模板
2. 技术决策和架构设计依赖个人经验
3. 缺乏系统的培训计划和知识共享机制
4. 过程改进缺乏系统性和数据支持
5. 跨团队协作流程不够明确

**三、过程改进计划**

基于当前Level 2现状，制定以下改进计划以达到Level 3：

**1. 过程标准化**

* 制定《SpringBoot+Vue开发规范》文档

包含代码规范、API设计规范、数据库设计规范

* 建立项目模板仓库

SpringBoot后端基础框架模板

Vue前端项目脚手架

统一的Docker部署配置

**2. 技术资产积累**

* 建立医院业务领域知识库

挂号业务流程文档

医疗行业数据标准

* 开发公共组件库

挂号排队算法组件

医疗时间处理工具类

通用患者信息表单组件

**3. 质量保障体系**

* 实施分层自动化测试
* 引入SonarQube进行静态代码分析
* 建立代码审查Checklist

**4. 度量与改进**

* 建立基本过程度量指标

| **指标名称** | **目标值** | **测量频率** |
| --- | --- | --- |
| 构建失败率 | <5% | 每日 |
| 代码覆盖率 | ≥70% | 每周 |
| 缺陷修复周期 | ≤2工作日 | 每周 |

* 每月举行过程改进会议

**5. 团队能力建设**

* 实施结对编程和代码走查
* 每月技术分享会(医疗IT专题)
* 建立新成员入职培训路径

**四、医院挂号系统具体改进措施**

针对原有系统的薄弱环节重点改进：

1. **需求工程**

引入需求追溯矩阵(RTM)

使用PlantUML绘制业务流程

1. **架构设计**

引入清晰的DDD分层架构：

├── application // 应用服务层

├── domain // 领域层

│ ├── model // 领域模型

│ └── service // 领域服务

├── infrastructure // 基础设施层

└── interfaces // 接口层

1. **异常处理**

建立统一的医疗业务异常体系，比如建立标准医疗错误码：DOCTOR\_NOT\_AVAILABLE

TIME\_SLOT\_OCCUPIED

PATIENT\_INFO\_INCOMPLETE

1. **性能优化**

挂号高峰期排队算法优化

引入Redis缓存热门科室号源

SQL查询性能调优