

CMMI 实施方案设计

CMMI 实施方案设计

- 1 实施框架
 - 1.1 改进模型等级
 - 1.2 过程机构及人员
 - 1.3 过程改进规程
 - 1.3.1 方针与目标
 - 1.3.2 定期评估和改进策划
 - 1.3.3 评审规程
 - 1.3.4 过程推广
 - 1.3.5 过程财富库管理
 - 1.3.6 所需培训和技能
 - 1.3.7 工具与设备要求
 - 1.3.8 风险识别
 - 1.4 过程改进计划
- 2 过程改进体系
 - 2.1 需求管理 (REQM)
 - 2.1.1 理解需求
 - 2.1.2 获得对需求的承诺
 - 2.1.3 管理需求变更
 - 2.1.4 维护需求的双向可追溯性
 - 2.1.5 确保项目产品与需求的一致性
 - 2.2 项目计划 (PP)
 - 2.2.1 开发人员培训计划
 - 2.2.2 质量保证计划
 - 2.2.3 测试计划
 - 2.2.3.1 计划的内容与范围
 - 2.2.3.2 单元测试的内容
 - 2.2.3.3 系统测试的内容
 - 2.3 项目监督和控制 (PMC)
 - 2.3.1 按计划监督项目
 - 2.3.2 管理纠正措施直至结束
 - 2.4 度量和分析 (MA)
 - 2.4.1 协调测量和分析活动
 - 2.4.2 提供测量结果
 - 2.5 配置管理
 - 2.6 过程和产品质量保证 (PPQA)
 - 2.7 供方协定管理

1 实施框架

1.1 改进模型等级

本次实施过程改进的机构为开发小组,而开发组项目成员为 5 人,固选用 CMMI2 级作为本次过程改进的模型。

1.2 过程机构及人员

- 发起者（Sponsor）：包括总经理和所有软件工程组成员，主要职责包括：
 - 从最上层开始推动 SPI
 - 支持过程改进，提供足够的资源、允许项目计划作适当调整
- 软件工程过程改进组组长（SEPG 组长）主要职责在于：
 - 制定 SPI 计划并实施
 - 组织 SEPG 组的工作，制定 OSSP
 - 新过程试用、评估，推动 OSSP 的实施，为工作组提供培训和支持
 - 维护过程数据库，阶段性评估软件过程，每月进行工作总结，并向发起者汇报工作进展
- 软件工程过程改进组

姓名	过程改进职责说明	项目角色
蔡佳伟	组长，负责推进和督导过程改进	研发部门经理
蔡佳伟	组员，负责项目监控PMC	项目经理
文豪	组员，负责项目规划PP，需求管理RM	项目经理
蔡云杉	组员，负责配置管理CM	美工
徐浩	组员，负责度量与分析MA	开发经理
魏宸、文豪	组员，负责过程与产品品质保证PPQA	测试经理
蔡云杉、魏宸	组员，负责供应商合约管理	公司执行经理

- 工作组

工作组是具体实施 CMMI 体系的项目组成员，应积极参与过程改进 SEPG 要对工作组的工作给予支持

- 软件过程改进顾问

中国软件评测中心

1.3 过程改进规程

1.3.1 方针与目标

- EPG 作为软件过程制定和优化的专业小组长期存在
- 系统集成和软件过程改进要结合市场特点和实践，有可操作性
- 注意工作阶段重点和工作的逐步完善
- 确定公司的过程改进计划
- 公司要执行的过程标准和产品标准

1.3.2 定期评估和改进策划

定期评估可以在年度管理发展计划中规定，而管理发展计划需要按照 CMMI 的评估模型开展公司的升级评估和内部评估。

改进策划需要从以下五个方面进行：

方面	说明
输入	以 ISO/CMMI 过程评估结果作为主要输入
时间	定义待改进活动的时间表
负责	提前规定负责每项活动的小组和个人
资源	提前确定要求的资源，包括资金、工具等
评审	评审首次发行和修改后的计划；需要软件经理、高级经理的评审和批准

1.3.3 评审规程

评审流程规定如下：

- 提交
 - 每个过程域计划文档都要提交给 QA
- 审计
 - QA 审计计划内容和格式，依据是产品审计检查单
- 评审
 - QA 审计通过后首先召开评审会，会签后生成评审记录
 - 若发生需求变更，需要提交项目组进行同行评审，得到所有成员的同意并由项目经理召开评审会，生成评审记录

1.3.4 过程推广

推广流程规定如下：

- 定义
 - 对过程体系文件和模板进行定义
- 试点
 - 选择试点项目，并运行已有体系
- 优化
 - 总结试点成果并优化已有体系
- 推广
 - 在推广项目中运用优化后的体系
- 迭代
 - 再次总结优化体系并更加广泛地推广

其中，每次优化，即过程改进的流程如下：

- 总结问题

- 对过程体系存在的问题进行总结
- 分析问题
 - 进行组会，对总结出的问题进行分析，提出更合适的过程体系
- 试用观察
 - 将提出的新体系试用于某项目，并由 QA 观察效果
- 循环改进
 - 若实施效果好，则扩大实施范围至整个研发部；若效果不好，则重复总结分析试用过程
- 记录结果
 - 记录过程改进的成果

还需注意的是，过程改进需要在执行完一个项目后、在开发中发现体系不适用或咨询师检查后给出更优建议的时候实施。以上三点即为过程改进的触发条件。

1.3.5 过程财富库管理

库名	存放内容	责任角色
标准过程库	小组标准过程集合	蔡佳伟
培训教材库	团队级培训教材及可共享的一些知识等资料	徐浩
经验库	根据公司在执行项目过程中形成的一些项目文档上的经验、技术解决问题经验、分析问题的解决经验、历史项目度量数据	文豪
过程改进库	存放团队在改进中的过程记录，包括改进计划、改进报告、组织测量数据、改进经验、案例等资料	魏宸
开发工具库	存放团队推荐使用的开发工具安装程序	蔡云杉
公司产品库	存放公司研发产品的全部技术文档，包括需求、设计、源代码、安装程序、使用手册等	蔡佳伟
组建库	存放一些项目级的可重用代码	文豪
评估库	存放评估准备和最终评估时的相关资料	蔡云杉、魏宸

1.3.6 所需培训和技能

培训内容：

针对需求，进一步加强软件开发规范和软件开发过程质量，了解客户的真正需求，提高开发人员开发素质，使开发人员成为高质量的软件工程师。结合项目进行案例讨论，根据讨论过程进行评估。通过上述培训，进一步提高开发人员技术水平，扩展开发人员的知识面和视野。

1.3.7 工具与设备要求

软硬件资源名称	级别	详细配置	获取时间或方式	使用说明
PyCharm	关键	无	官网获取	开发主要使用的 IDE
JS, SQL, Django, Vue	关键	无	无	使用 Django 作为后端, 数据库管理使用 SQL 语言
Edge	关键	版本: 130.0.2849.80 (正式版本) (64 位)	无	开发阶段用于调试代码

1.3.8 风险识别

目前识别的风险主要体现在四个方面：需求、技术、成本和进度。详细介绍有如下几类：

- 需求风险
 - 需求已经成为项目基准，但需求还在继续变化
 - 需求定义欠佳，而进一步的定义会扩展项目范畴
 - 产品定义含混的部分比预期需要更多的时间
 - 在做需求中客户参与不够
 - 缺少有效的需求变化管理过程
- 计划编制风险
 - 计划、资源和产品定义全凭客户或上层领导口头指令，并且不完全一致
 - 计划是优化的，是“最佳状态”，但计划不现实，只能算是“期望状态”
 - 产品规模（代码行数、功能点、与前一产品规模的百分比）比估计的要大
 - 完成目标日期提前，但没有相应地调整产品范围或可用资源
 - 涉足不熟悉的产品领域，花费在设计和实现上的时间比预期的要多
- 组织和管理风险
 - 仅由管理层或市场人员进行技术决策，导致计划进度缓慢，计划时间延长
 - 低效的项目组结构降低生产率
 - 管理层审查 决策的周期比预期的时间长
 - 预算削减，打乱项目计划
 - 管理层作出了打击项目组织积极性的决定
 - 缺乏必要的规范，导致工作失误与重复工作
 - 非技术的第三方的工作（预算批准、设备采购批准、法律方面的审查、安全保证等）时间比预期的延长
- 人员风险
 - 作为先决条件的任务（如培训及其他项目）不能按时完成
 - 开发人员和管理层之间关系不佳，导致决策缓慢，影响全局
 - 缺乏激励措施，士气低下，降低了生产能力

- 某些人员需要更多的时间适应还不熟悉的软件工具和环境
- 项目后期加入新的开发人员，需进行培训并逐渐与现有成员沟通，从而使现有成员的工作效率降低
- 由于项目组成员之间发生冲突，导致沟通不畅、设计欠佳、接口出现错误和额外的重复工作
- 不适应工作的成员没有调离项目组，影响了项目组其他成员的积极性
- 没有找到项目急需的具有特定技能的人
- 开发环境风险
 - 设施未及时到位
 - 设施虽到位，但不配套，如没有电话、网线、办公用品等
 - 设施拥挤、杂乱或者破损
 - 开发工具未及时到位
 - 开发工具不如期望的那样有效，开发人员需要时间创建工作环境或者切换新的工具
 - 新的开发工具的学习期比预期的长，内容繁多
- 用户风险
 - 用户对于最后交付的产品不满意，要求重新设计和重做
 - 用户的意见未被采纳，造成产品最终无法满足用户要求，因而必须重做
 - 用户对规划、原型和规格的审核决策周期比预期的要长
 - 用户没有或不能参与规划、原型和规格阶段的审核，导致需求不稳定和产品生产周期的变更
 - 用户答复的时间（如回答或澄清与需求相关问题的时间）比预期长
 - 用户提供的组件质量欠佳，导致额外的测试、设计和集成工作，以及额外的用户关系管理工作
- 产品风险
 - 矫正质量低下的不可接受的产品，需要比预期更多的测试、设计和实现工作
 - 开发额外的不需要的功能（镀金），延长了计划进度
 - 严格要求与现有系统兼容，需要进行比预期更多的测试、设计和实现工作
 - 要求与其他系统或不受本项目组控制的系统相连，导致无法预料的设计、实现和测试工作
 - 在不熟悉或未经检验的软件和硬件环境中运行所产生的未预料到的问题
 - 开发一种全新的模块将比预期花费更长的时间
 - 依赖正在开发中的技术将延长计划进度
- 设计和实现风险
 - 设计质量低下，导致重复设计
 - 一些必要的功能无法使用现有的代码和库实现，开发人员必须使用新的库或者自行开发新的功能
 - 代码和库质量低下，导致需要进行额外的测试，修正错误，或重新制作
 - 过高估计了增强型工具对计划进度的节省量
 - 分别开发的模块无法有效集成，需要重新设计或制作
- 过程风险
 - 大量的纸面工作导致进程比预期的慢
 - 前期的质量保证行为不真实，导致后期的重复工作
 - 太不正规（缺乏对软件开发策略和标准的遵循），导致沟通不足，质量欠佳，甚至需重新开发

- 过于正规（教条地坚持软件开发策略和标准），导致过多耗时于无用的工作
- 向管理层撰写进程报告占用开发人员的时间比预期的多
- 风险管理粗心，导致未能发现重大的项目风险

1.4 过程改进计划

里程碑	活动	参与者	成功标准
1	建立符合 CMMI2 即标准的可操作的过程文件	项目小组成员	建立符合 2 级标准的可操作过程文件，并通过评审
2	选择试点项目推行过程体系，指定过程域的负责人，启动试点项目	项目小组成员	各过程域有指定的负责人，试点项目
3	试点项目总结	项目小组成员	总结 CMMI 体系在试点中的推广效果，提取出好的经验和不足之处
4	评审修正后的过程体系	项目小组成员	建立起更优化的体系
5	启动推广项目，正式应用过程体系	项目小组成员	推广项目建立优化后的过程体系，制定各过程域责任人
6	预评估	项目小组成员、老师、助教	识别到项目前体系和 CMMI2 级标准的偏差，并进一步优化过程体系
7	正式评估	项目小组成员、老师、助教	通过正式评估的结果进一步优化过程体系

2 过程改进体系

2.1 需求管理（REQM）

CMMI-DEV 的需求管理过程需要我们实现一个特定目标（SG），即要求我们管理工程的所有需求并且能够识别出需求、项目计划与工作产品之间的一致性并作出相应处理，以确保需求、项目计划与工作产品之间的一致性。根据 CMMI 规范，我们需要进行如下 5 个特定方法（SP）：

2.1.1 理解需求

需求管理的目标需要我们与需求提供者明确需求的意义。我们需要确保能够找到一个合适的需求提供方，并且确保需求接收方与需求提供方对需求有着一致的理解。我们需要如下方法完成此要求：

- 需求提供方评估方案，用于筛选合适的需求提供方。
- 需求评估方案，用于筛选需求提供方提供的需求。
- 需求总结，根据上述方案对提供方提供的需求进行分析，筛选不一致的、无法实现的需求，并对能够接受的需求予以确认。

2.1.2 获得对需求的承诺

在得到能够接受的需求后，我们需要令进行实现具体需求的人员认识并理解需求及其变化。我们需要需求评估确认书，确定需求的优先级、重要性并令人员进行承诺与确认。

2.1.3 管理需求变更

在项目推进过程中，由于种种原因，需求的内容很有可能发生变更。因此我们需要一套完整的流程去管理需求的变更。我们需要确定如下方法：

- 需求变更请求书，用于规范化需求变更请求的形式。
- 需求变更评估方案，用于确定某一个需求变更的请求对项目的影响。
- 需求数据库，用于跟踪需求变化的历史并确定当前的需求状态。

2.1.4 维护需求的双向可追溯性

本方法要求我们的项目能够维护产品与需求之间的双向可追溯性，并保证底层需求能够追溯到一个有效的源头。我们需要系统性地追踪需求的变化以确保此方法的正确实现。

2.1.5 确保项目产品与需求的一致性

我们需要有能力发现项目产品与需求间的不一致性，并在发现此问题时及时采取对应的行动以进行解决。

2.2 项目计划（PP）

2.2.1 开发人员培训计划

本程序由于涉及到 MySQL、Django、Vue.js 等框架的使用，因此有必要对开发人员进行相关开发技能的培训。

其中，对前端开发人员应当进行 4~6 天的 Vue.js 及其常用开发框架的开发培训；对后端开发人员应当分别进行 1~3 天的 MySQL 接口开发培训，以及 Python、Django 的开发培训。

2.2.2 质量保证计划

项目应当进行完善的质量保证措施，严格按照 2.1 节要求进行需求管理，确保工作产品能够符合项目需求。质量保证相关者应当订立明确清楚的质量保证计划，并按照计划进行相应工作。

质量保证人员应当定期、按阶段地对上一阶段的工作过程及工作成果进行评估，并根据评估成果采取相应措施以便改善质量保证的成效。

2.2.3 测试计划

2.2.3.1 计划的内容与范围

随着本项目的推进，我们需要在不同范围、不同粒度上对本项目进行测试。

从软件开发过程而言，我们需要在每个模块开发完成之后，按照此模块的功能需求对此模块进行**单元测试**，并在各个模块开发完成后对其进行**集成测试**；在系统整体完成开发后，我们需要在整个项目范围内进行**系统测试**，模拟正常使用的情景，确保项目程序的可用性。

从测试方法而言，我们在上述各个过程中需要分别进行**黑盒测试**、**白盒测试**。此外，我们还需要对代码进行静态分析，确保代码风格的一贯性、代码编写的规范性与一致性。

2.2.3.2 单元测试的内容

目前而言，本项目需要对以下内容进行单元测试：

- 前端界面的可用性
- 一般搜索功能
- 高级搜索功能
- 对话功能
- 后台管理功能

应当在每个模块开发结束后对对应的模块进行不同的测试。每个测试应当能够确保完整考虑到不同的输入取值范围及可能的结果。

在黑盒测试中，我们应当确保测试输入的广泛性，确保各项输出与预期输出相一致。

在白盒测试中，我们应当使得测试输入能够覆盖代码的每一个分支，确保代码能在全部状况下按照符合期待的方式运行。

2.2.3.3 系统测试的内容

在单元测试通过后，我们需要接着完成系统测试。我们需要模拟正常使用的环境，对整个系统进行不同的操作，并验证其是否产生了预期的结果。

2.3 项目监督和控制（PMC）

项目监督和控制（Process Monitoring and Control）过程要求我们实现 2 个特定目标，分别为：按计划监督项目、管理纠正措施直至结束。

2.3.1 按计划监督项目

此目标要求我们按计划监督项目的进展与效率。其被分为以下7个子目标：

- **监督项目计划的参数。**即要求我们对计划、预算等参数的计划过程及效率的因数进行监督。我们需要进行项目效率记录、显著偏差记录、效益记录。
- **监督承诺。**我们需要对项目计划中的承诺的进行状况进行监督，并形成记录。
- **监督项目风险。**我们需要对项目的风险进行监督并做出记录。
- **监督数据管理。**我们需要对数据管理情况进行监督，并按照需求的变更做出相应调整，并进行记录。
- **监督项目关系者参与。**我们需要将项目关系者（stakeholder）的参与情况进行监督，确保其参与是合理的，并在必要的情况下重新考虑项目计划。此项同样需要我们生成记录。
- **进行进度管理。**我们需要时刻关注项目的进度，取得的进展及结果，并形成项目进展报告。
- **进行里程碑管理。**我们需要设立项目的进展里程碑目标，并在项目的对应时刻进行里程碑评价结果的记录。

2.3.2 管理纠正措施直至结束

我们需要在项目进展过程中项目的效率或结果与计划产生显著差异时不断管理纠正措施。具体措施如下：

- **分析原因：**我们需要在出现问题时分析其原因并制定纠正措施，制定纠正措施表。
- **实行措施：**由上述原因产生对应的实施计划并实施之。
- **管理纠正措施：**我们需要跟踪上述纠正措施的结果，直到结束。

2.4 度量和分析 (MA)

这一区域需要我们开发并维护一个度量方法来满足管理的需要。我们需要确定度量分析的目标，令其与项目要求保持一致；确定这一系列环节所需要的技术机制并实施之；依据这些度量产生合适的纠正措施，为将来的测量提供基础。

分为如下两个目标：

2.4.1 协调测量和分析活动

我们需要按照指定的信息需求和目标协调测量与分析活动。

- 我们需要确立度量的目标，并明确每个度量目标所对应的措施；
- 我们需要确定不同度量目标所对应的具体措施；
- 我们需要确定如何获取并储存数据。
- 我们需要制定分析的具体步骤。

2.4.2 提供测量结果

我们需要提供按照上述流程产生的测量结果。步骤如下：

- 获取测量数据。工作产品是结果数据集。
- 分析测量数据。工作产品是数据分析报告。
- 存储数据和结果。工作产品是数据存储。
- 与相关人员就结果进行沟通。工作产品是沟通用的分析报告与上下文信息。

2.5 配置管理

配置管理 (CM) 要求我们进行配置的识别、控制、记录、调整，以保证工作产品的完整性。其包括 3 个目标：建立基线、跟踪和控制更改、建立完整性。

建立基线要求我们：

- 识别配置项，即识别出需要被置于配置管理之下的配置项目、组件和相关的工作产品。
- 建立配置管理系统，产生受控的工作产品及访问控制过程等。
- 创建或释放用于内部或顾客的基线。

跟踪和控制更改要求我们：

- 跟踪更改，即跟踪控制项的更改。控制项的更改请求需要分析以便评估对工作产品的影响。
- 控制控制项的变化，追踪修改历史，并归档基线。

建立完整性要求我们：

- 建立配置管理记录；
- 进行配置审计，以便维护配置基线的完整性。

2.6 过程和产品质量保证 (PPQA)

过程和产品质量保证用于向工作和管理人员提供一个客观的视角来观察工作流程和相关产品。为达成此区域，我们需要达成两个目标：客观地评估过程和工作产品、提供客观的洞察。

前者要求我们实现：

- 客观地评估过程，要求我们形成评估、不兼容报告及相应的纠正过程。

- 客观评价工作产品，工作产品同上。

后者要求我们实现：

- 沟通并解决不兼容问题，产生纠正措施报告、评估报告、质量趋势分析；
- 创建相关记录。

2.7 供方协定管理

供方协定管理（SAM）用于管理从供方获得的产品和服务。获得的产品和服务包括子系统、软硬件、文档、物件等。为了满足本需要，我们需要：

- 建立供方协议。我们需要判断获取类型、选择供应商、建立供方协议。
- 满足供方协议。我们需要执行供方协议、接受获取的产品、确保产品过渡顺利。