

江苏省软件行业协会标准规范

T/JSIA xxx-2026

基于大语言模型的软件开发规范 (征求意见稿)

2026 - X - X 发布

2026 - X - X 实施

目 次

目 次.....	2
前 言.....	3
基于大语言模型的软件开发规范.....	4
1 范围.....	4
2 规范性引用文件.....	4
3 术语和定义.....	4
4 核心原则.....	5
5 大语言模型选型.....	5
6 基于大语言模型的软件开发.....	6
6.1 总体流程.....	6
6.2 软件开发各阶段要求.....	6
6.2.1 AI 构建原型.....	6
6.2.2 AI 架构建模.....	6
6.2.3 AI 功能建模.....	7
6.2.4 AI 库表建模.....	7
6.2.5 AI 编程集成.....	7
7 基于大语言模型的系统迭代.....	7
7.1 总体流程.....	7
7.2 系统迭代各阶段要求.....	8
7.2.1 AI 分析跨系统影响范围.....	8
7.2.2 AI 分析 SPEC 增量.....	8
7.2.3 AI 迭代 SPEC 模型.....	8
7.2.4 AI 编程集成.....	8
8 质量与安全管控.....	9
9 知识产权管理.....	9
10 实施与应用.....	9

前 言

本规范依据GB/T 1.1-2020给出的规则起草。

本规范由江苏省软件行业协会提出和归口。

本标准起草单位：****。

本标准主要起草人：

本标准为首次发布。

基于大语言模型的软件开发规范

1 范围

本规范规定了基于大语言模型的软件开发流程、系统迭代、质量管控、知识产权、实施应用等核心内容。

本规范适用于专业软件企业、大型企业软件研发部门、大型企业数科公司等，基于大语言模型开展的软件开发全流程活动，涵盖需求分析、软件建模、代码生成、系统测试、系统迭代等关键环节。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。其中，凡是注日期的引用文件，仅注日期的版本适用于本文件；凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 31102-2025 系统与软件工程 软件工程知识体系

GB/T 45802-2025 系统与软件工程 生存周期过程 需求工程

《中华人民共和国数据安全法》

江苏省软件行业协会团体标准知识产权管理规则

开源软件合规使用通用指南

3 术语和定义

1、“软件 SPEC 模型+AI 大模型”双模驱动架构

以 AI 大模型为核心能力、以软件 SPEC 模型作为核心资产，由 AI 大模型将业务需求持续迭代转化成高质量的软件 SPEC 模型，再由 AI 大模型根据该 SPEC 模型生成高质量源码，并完成自动化测试。

2、大语言模型(LLM)

使用大量文本数据训练的深度学习模型，该模型具备自然语言理解、逻辑推理、代码生成、交互对话等能力，可支撑软件开发全流程任务。

3、软件 SPEC 模型

以软件规格说明为核心的标准化软件模型，完整定义软件功能、界面、数据、行为及

验证规则，包含功能结构、功能模块、用户界面、数据模型、集成用例等核心要素，用于驱动大语言模型进行软件开发与迭代的核​​心依据。

4、软件 SPEC 基线

经评审确认、正式发布的软件 SPEC 模型版本，作为后续开发、变更、测试的基准依据，不可随意修改。

5、软件源码基线

在软件开发过程中，对源代码库在某一特定时间点的稳定版本所作的快照。该版本经过测试、评审并确认为稳定可靠，可作为后续开发、测试或发布的基础。

6、Harness 工程

支撑 AI 自动化编程、集成、测试的可定制化工艺流程，确保 AI 能够满足不同技术环境和技术规范的要求而产出高质量代码。

4 核心原则

1、科学性

遵循软件工程与大语言模型技术规律，构建全流程标准化、体系化的开发规范。

2、可落地性

贴合国内软件产业实际，流程可直接应用于软件研发场景，团队可快速上手。

3、安全性

严格管控代码质量、数据安全与知识产权，规避技术与合规风险。

4、高效性

通过大语言模型能力，实现软件研发团队稳定提效、降低人工成本。

5、合规性

遵守国家法律法规、行业监管要求及开源合规规则，全程合法合规。

5 大语言模型选型

1、能力适配

具备自然语言理解、逻辑推理、代码生成、测试用例生成、交互对话等能力。

2、合规安全

通过国家网信办算法备案，无数据外泄、非法采集风险。

3、可控性

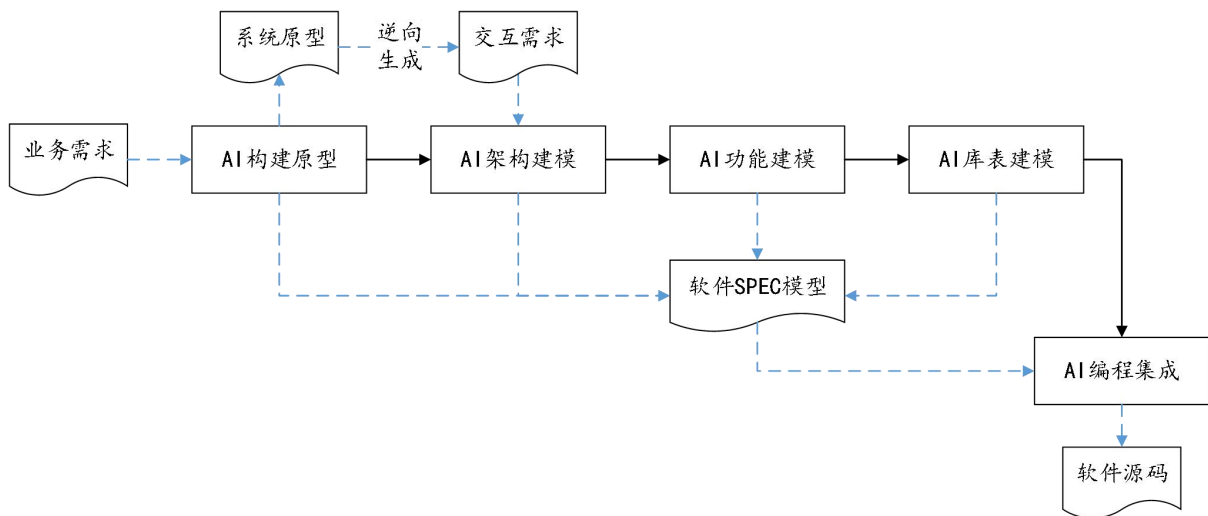
支持公有云调用。

支持私有化部署和本地调用，可配置权限、日志、审计。

6 基于大语言模型的软件开发

6.1 总体流程

整个开发过程以生产、迭代和应用软件 SPEC 模型为主线，涵盖 AI 构建原型、AI 架构建模、AI 功能建模、AI 库表建模、AI 编程集成五个核心阶段（见下图），实现 SPEC 模型驱动的工程化 AI 应用。



6.2 软件开发各阶段要求

6.2.1 AI 构建原型

- 1) 依据业务需求文档，通过大语言模型生成 HTML 格式可交互系统原型。
- 2) 与业务方确认原型，通过大语言模型快速迭代系统原型，并达成共识。
- 3) 基于确认的系统原型，逆向生成交互需求文档。
- 4) 系统原型需满足业务场景验证。

6.2.2 AI 架构建模

- 1) 结合业务需求、交互需求，通过大语言模型完成整体架构建模，人工审核修订。
- 2) 建模内容包括用户角色、系统概述、质量需求、功能域划分、模块边界。
- 3) 整体架构建模内容，需覆盖全部业务需求。

6.2.3 AI 功能建模

- 1) 通过大语言模型按功能模块逐个完成功能建模，人工审核修订。
- 2) 功能建模包括功能模块建模、用户界面建模、集成用例建模。
- 3) 功能模块需明确用户场景、数据对象、数据实体、业务规则、功能点。
- 4) 用户界面需明确界面规则、字段、按钮、界面事件能力，定义界面事件能力与功能点之间的关联调用关系。
- 5) 集成用例需明确人机操作步骤。
- 6) 无需求遗漏、逻辑自洽。

6.2.4 AI 库表建模

- 1) 通过大语言模型，按功能模块逐个完成数据表建模，人工审核修订。
- 2) 数据表建模需精准定义数据表名称、字段名称、字段类型、长度、默认值等全部信息。
- 3) 数据表符合数据规范、无结构冲突。

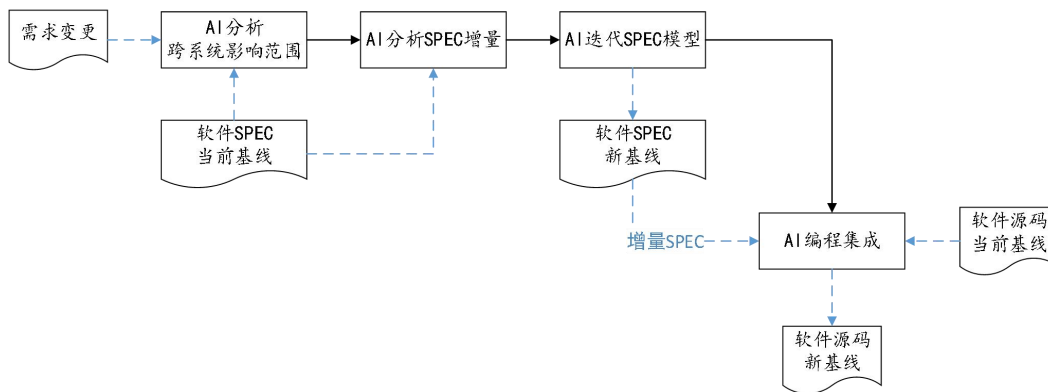
6.2.5 AI 编程集成

- 1) 基于软件 SPEC 模型，通过大语言模型自动生成符合企业规范的源代码，完成通用组件集成、依赖配置、环境适配。
- 2) 依据集成用例，通过大语言模型执行自动化测试，验证功能符合性。
- 3) 依托 Harness 工程保障，确保大语言模型能够满足不同技术环境和技术规范的要求而产出高质量代码。
- 4) 软件源码语法合规、功能可用、通过自动化测试。

7 基于大语言模型的系统迭代

7.1 总体流程

整个迭代过程涵盖 AI 分析跨系统影响范围、AI 分析 SPEC 增量、AI 迭代 SPEC 模型、AI 编程集成四个核心阶段（见下图），大幅降低变更遗漏风险，实现高效迭代软件源码。



7.2 系统迭代各阶段要求

7.2.1 AI 分析跨系统影响范围

- 1) 根据需求变更，基于软件 SPEC 当前基线，通过大语言模型自动分析变更对关联系统、模块、数据的影响。
- 2) 形成影响范围分析任务报告，明确涉及的软件项目、功能、变更说明等内容。

7.2.2 AI 分析 SPEC 增量

- 1) 针对变更涉及的软件项目，由大语言模型识别新需求并生成 SPEC 迭代单，人工审核修订。
- 2) SPEC 迭代单中需详细明确新增功能点与变更功能点。

7.2.3 AI 迭代 SPEC 模型

- 1) 根据 SPEC 迭代单，大语言模型迭代软件 SPEC 模型。
- 2) 人工审核修订，发布生成软件 SPEC 新基线

7.2.4 AI 编程集成

- 1) 依据增量 SPEC，通过大语言模型自动迭代符合规范的软件源码。
- 2) 根据集成用例，通过大语言模型执行自动化测试，验证代码功能。
- 3) 发布软件源码新基线。
- 4) 软件源码语法合规、功能可用、通过自动化测试。

8 质量与安全管控

1、代码质量校验

对大语言模型生成的代码，进行语法、逻辑、规范符合性校验。

2、Harness 工程保障

包括 AI 编写业务功能代码工艺流程、集成企业组件工艺流程(如审批引擎、权限管理等)、自动化测试工艺流程，确保 AI 能够满足不同技术环境和技术规范的要求而产出高质量代码。

9 知识产权管理

1、权属界定

严格遵守知识产权法律法规，不侵犯第三方专利、著作权、商业秘密。

2、AI 生成代码权属

遵循“谁主导开发、谁享有权利”的原则。

3、开源合规

AI 生成代码中如使用开源组件，确保符合开源使用条件，明确标注开源组件来源。

10 实施与应用

1、企业落地要求

参照本规范搭建基于大语言模型的研发流程，配套制度、工具、培训。

建立 SPEC 模型、代码基线、变更管控的标准化管理体系。

2、监督与迭代

江苏省软件行业协会负责规范宣贯、培训、监督、评估。

结合大模型技术发展与产业实践，定期迭代规范版本。

3、推广与评价

鼓励企业开展规范落地试点，形成典型案例，纳入行业优秀实践推广。