

江苏省软件行业协会标准规范

T/JSIA xxx-2026

面向 AI 编程的软件 SPEC 编制规范 (征求意见稿)

2026 - X - X 发布

2026 - X - X 实施

目 次

目 次.....	2
前 言.....	3
面向 AI 编程的软件 SPEC 编制规范.....	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.....	4
5 软件 SPEC 核心构成要素.....	5
5.1 重要规则.....	5
5.2 功能模型.....	5
5.3 数据模型.....	6
5.4 入口模型.....	7
5.5 接口模型.....	7
5.6 用例模型.....	7
5.7 库表模型.....	7
6 软件 SPEC 结构化模板.....	8
6.1 重要规则.....	8
6.2 功能模型.....	9
6.2.1 功能结构.....	9
6.2.2 功能模块.....	9
6.2.3 用户界面.....	10
6.3 数据模型.....	10
6.4 入口模型.....	10
6.5 接口模型.....	11
6.6 用例模型.....	11
6.7 库表模型.....	12
7 知识产权管理.....	12
8 实施与应用.....	13

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：

本标准首次发布。

面向 AI 编程的软件 SPEC 编制规范

1 范围

本规范定义了面向 AI 编程场景下，软件 SPEC 的术语与定义、核心构成要素及结构化模板。

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 AI 编程活动。

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 9385-2008 计算机软件需求规格说明规范

GB/T 8567-2006 计算机软件文档编制规范

江苏省软件行业协会团体标准知识产权管理规则

3 术语和定义

1、软件 SPEC (Software Specification)

面向 AI 编程时，用于描述软件系统需求、设计、约束及行为的、可被 AI 模型解析的结构化完整技术文档集合。

2、SPEC 驱动开发 Specification-Driven Development (SDD)

以明确、结构化、可验证的规范 (Specification) 作为软件开发核心驱动力的工程方法论。该方法强调在编写代码之前，通过“规范”对齐人与 AI、团队成员之间的系统意图，从而提升代码质量、可维护性和协作效率。

4 核心原则

面向 AI 编程的软件 SPEC 编制应遵循以下核心原则，确保 SPEC 可被 AI 准确解析并生成高质量代码。

1、确定性原则

SPEC 中每一条需求、规则、约束必须表达为可验证真假的确定性命题。禁止使用模糊词汇如“等”、“可能的”、“适当的”、“良好的”等。

2、结构化表达原则

遵循“一个逻辑在一处定义，多处引用”的原则，避免在多处重复描述。

3、可追溯性原则

每个功能模块、用户界面、数据库表、接口、用例都必须具有唯一标识符。SPEC 变更必须记录变更原因、影响范围，并保持与代码基线的双向追溯。

4、原子化分解原则

SPEC 中的每个功能点、数据项或规则描述应达到适当的分解层次，其颗粒度应与代码结构形成显性映射。

5 软件 SPEC 核心构成要素

5.1 重要规则

用于约束 AI 对系统全局观和基础认知，包括以下内容：

1、系统概述

对系统的全面、结构化描述，包括系统主要作用、同类系统、核心场景、亮点功能、技术架构及所关联的外围系统。

2、用户角色

描述各类用户角色，明确其核心职责与权限范围。

3、质量需求

涵盖性能、安全性、可靠性、易用性、可维护性及合规性等非功能性要求。

4、建模技能

包括通用模块、通用界面、界面模板及界面规范等可复用资产。

5、领域知识

描述业务领域的核心概念、术语表以及不可违背的业务规则。

5.2 功能模型

1、功能结构

功能结构由层次化的功能域构成，功能域基于业务能力进行稳定分组。对每个末级功能域，描述其功能描述，并明确所包含的数据实体及关键数据项。

2、功能模块

功能模块是软件系统中边界清晰、功能内聚的逻辑单元，由一组紧密相关的用户场景、业务规则、功能点组成。

1) 用户场景

为达成特定业务目标，用户与系统进行交互的过程，需明确用户角色、执行操作序列及执行目标。

2) 业务规则

功能模块内必须遵循的业务逻辑约束、计算规则或决策条件，每条规则应表达为确定性命题。

3) 功能点

具有明确的业务数据处理规则和校验标准、可独立执行的最小颗粒度功能单元。

3、用户界面

1) 界面规则

描述界面控件、及其类型和交互控制规则，如可见性、可编辑性、校验规则等。

2) 界面事件能力

描述用户操作（如查询、保存）所触发的处理逻辑，包括前置条件、调用的功能点/外联接口、正常处理路径以及异常处理路径。

5.3 数据模型

用于统一描述系统所涉及的数据结构，确保数据描述清晰、关系明确、可被 AI 准确解析与实现。

1、业务对象

系统中具有独立业务含义的数据集合，通常对应现实世界中的业务单据、事件记录或参照信息。

2、数据实体

构成业务对象的具体组成单元，通常对应业务单据的主记录、子记录等。

3、业务对象与数据实体的关系

1 类业务对象包括 1 到 N 类数据实体，和 0 到 N 类其他业务对象。

5.4 入口模型

用于描述用户通过不同终端进入系统的导航路径，明确各终端下的菜单结构与用户界面的映射关系，确保 AI 能够准确生成多端路由逻辑。

1、入口结构

按终端类型（如 PC 端 Web、移动端 H5/App、平板端等）分别描述系统的多级菜单树，每个终端类型的菜单树独立描述，采用树形结构组织。

2、末级入口

末级入口是指入口结构中无子菜单的叶子节点。每个末级入口必须映射到一个明确的用户界面，映射关系应使用确定性语言描述，禁止使用“等”“可能”等模糊词汇。

5.5 接口模型

用于描述目标系统与外联系统之间的所有交互边界，明确外联系统提供的接口描述，确保 AI 能够准确生成接口调用代码。

1、外联系统

描述与本系统存在数据交换的第三方系统。

2、外联接口

描述每个外联系统中与本系统相关的具体接口，包括接口名称、请求参数、返回参数。

5.6 用例模型

从用户视角描述系统应提供的业务价值，明确参与者与系统的交互流程，并支持跨功能模块、跨系统的端到端业务场景验证。

1、用例结构

描述系统中所有用例的组织方式，按功能域或业务主题进行分组。

2、集成用例

描述多个功能模块或外部系统协作完成的端到端业务流程，涉及多个用户角色或系统间的交互序列。集成用例用于验证跨模块、跨系统的整体业务价值。

5.7 库表模型

用于描述系统的持久化存储结构，明确数据库、库表域及数据表信息。

1、数据库

根据系统的业务划分、数据特性或性能要求，将一个系统划分为多个不同的数据库，每个数据库承担不同的存储职责（如主业务库、日志库、缓存库、配置库等）。

2、库表域结构

库表域结构是数据表的逻辑分组，通常对应功能域或业务域。在每个数据库内部，可按模块或业务进一步划分库表域。

3、数据表

数据表是持久化存储的具体实现，与数据实体相对应。每个数据表归属于某个数据库下的某个库表域。

6 软件 SPEC 结构化模板

本模板为面向 AI 编程的软件 SPEC 的标准结构。编制 SPEC 时，按此模板组织文档，并遵循确定性、结构化表达、可追溯性、原子化分解原则。

6.1 重要规则

1、系统概述

1) 主要作用：系统存在的基本价值，概括系统解决的核心问题、服务的目标用户和创造的根本价值。

2) 同类系统：与目标系统在功能领域、业务模式或技术方案上相似的知名或典型系统/产品，用于类比和定位。

3) 核心场景：系统最典型、最频繁的用户使用场景或业务流程，描述用户角色在特定情境下的关键操作。

4) 亮点功能：系统区别于普通同类系统的、具有创新性或显著竞争力的功能特性。

5) 技术架构：系统采用的技术栈、架构风格和关键组件，包括前端、后端、数据库、部署环境等。

6) 外围系统：需要与目标系统进行数据交换或服务调用的外部系统、第三方服务或平台。

2、用户角色

描述每个用户角色信息，包括角色代码、角色名称、核心职责、权限范围。

3、质量需求

1) 性能：如批量处理能力、响应时间、吞吐量要求

- 2) 安全性：如访问控制、认证与集成、数据安全要求
- 3) 可靠性：如可用性、容错性要求
- 4) 易用性：如界面与操作的便捷要求
- 5) 可维护性：如配置灵活性要求
- 6) 合规性：如监管合规要求

4、建模技能

- 1) 通用模块：模块名称及路径
- 2) 通用界面：界面名称及组件库
- 3) 界面模板：模板名称及使用场景
- 4) 界面规范：布局、配色、字体、间距等

5、领域知识

- 1) 核心概念：概念名称及描述
- 2) 术语表：术语名称及描述
- 3) 不可违背的业务规则：包括规则名称、确定性规则描述

6.2 功能模型

6.2.1 功能结构

功能域	功能域级别	末级功能域	功能域描述
功能域名称	1		
功能域名称	2	是	功能域代码、功能域概述、各数据实体及关键数据项
功能域名称	2	是	功能域代码、功能域概述、各数据实体及关键数据项
功能域名称	1		
功能域名称	2		
功能域名称	3	是	功能域代码、功能域概述、各数据实体及关键数据项
功能域名称	3	是	功能域代码、功能域概述、各数据实体及关键数据项
功能域名称	2		

6.2.2 功能模块

对每个功能模块，按以下结构描述：

1) 功能模块代码

2) 功能模块名称

3) 所属功能域

4) 模块描述

5) 用户场景：描述每个用户场景，包括场景标题、用户角色、场景描述、执行操作序列、每个操作序列所调用的功能点

6) 业务规则：描述每个业务规则，包括规则标题、确定性的规则内容。

7) 功能点：描述每个功能点，包括功能点代码、功能点类型（交互类/接口类/自动类/基础类）、功能点简述、功能点验收标准；其中功能点验收标准包含应遵守的规则、数据处理目标、需调用的其他功能点。

6.2.3 用户界面

对每个用户界面，按以下结构描述：

1、界面规则

描述界面上的每个控件（如按钮、列表、字段、查询条件），包括控件名称、控件类型、可见性条件、可编辑性条件。

2、界面事件能力

描述每个界面事件能力的处理逻辑，包含应遵守的规则、调用的功能模块功能点、成功处理路径、异常处理路径。

6.3 数据模型

业务对象与数据实体的描述归属于各功能模块。对每个业务对象和数据实体，按以下结构描述：

1、业务对象

描述业务对象的信息，包括业务对象名称、所包含的数据实体及与这些实体的关系、所包含的其他业务对象及与这些对象的关系。

2、数据实体

描述每个数据实体的信息，包括编号、名称、关键数据项、以及对应的数据表。

6.4 入口模型

描述不同终端（如 PC 端 Web、移动端 H5/App、平板端等）进入系统的导航路径，明确菜单结构与用户界面的映射关系。每个终端按以下结构描述。

菜单名称	菜单级别	末级入口	映射界面
菜单名称	1		
菜单名称	2	是	用户界面
菜单名称	2	是	用户界面
菜单名称	2	是	用户界面
菜单名称	1		
菜单名称	2	是	用户界面
菜单名称	2	是	用户界面

6.5 接口模型

对每个外联系统及外联接口，按以下结构描述：

1、外联系统

描述每个外联系统的信息，包括外联系统名称、外联系统域名。

2、外联接口

描述每个外联接口的信息，包括所属外联系统、接口编号、接口名称、接口描述、请求参数、返回参数。

6.6 用例模型

1、用例结构

用户域	用例域级别
用例域名称	1
用例域名称	2
用例域名称	2
用例域名称	1
用例域名称	2
用例域名称	2
用例域名称	2

2、集成用例

对每个集成用例，按以下结构描述：

1) 集成用例代码

2) 集成用例名称

3) 所属用例域

4) 集成用例描述

5) 用户场景描述：在每个用户场景下，描述用户与系统的操作序列、以及每步操作序列中的用户操作和系统响应。

6.7 库表模型

1、数据库

根据系统需要，划分多个不同数据库，分别承担不同的职责。描述每个数据库的信息，包括：数据库编码、数据库名称、数据库类型（如 MySQL、PostgreSQL 等）、职责说明。

2、库表域结构

在每个数据库下描述库表域结构。

库表域	库表域级别
库表域名称	1
库表域名称	2
库表域名称	2
库表域名称	1
库表域名称	2
库表域名称	2
库表域名称	2

3、数据表

对每个数据表，按以下结构描述：

1) 数据表代码

2) 数据表名称

3) 所属数据库

4) 所属库表域

5) 数据表字段：描述每个字段的代码、字段名、类型、长度、默认值、是否主键、是否引用表外键。如引用表外键，则需描述所引用的数据表及对应字段。

7 知识产权管理

1、权属界定

严格遵守知识产权相关法律法规，确保不侵犯第三方的专利权、著作权及商业秘密；依据合同约定明确 SPEC 文档著作权归属。

2、SPEC 文档版权

本规范定义的 SPEC 文档内容（包括模板、示例及结构）的版权归起草单位所有。企业依据本规范编制的具体项目 SPEC，其版权由编制方或合同约定方享有。

3、第三方代码与开源许可

若 SPEC 中引用了第三方库、开源组件或外部接口，必须明确其许可协议，并确保 AI 生成的代码符合相应许可的使用、修改与分发要求。

4、保密义务

参与 SPEC 编制与 AI 编程的相关人员，对涉及商业秘密、未公开的技术方案及敏感数据负有保密责任。SPEC 文档中若包含敏感信息，应标注密级并采取访问控制措施。

5、侵权责任

因 SPEC 描述不当（如模糊、歧义或遗漏必要约束）导致 AI 生成的代码侵犯第三方知识产权的，责任由 SPEC 提供方承担。因 AI 模型训练数据或生成内容引发的侵权问题，依据服务协议及适用法律另行界定。

8 实施与应用

1、企业落地要求

参照本规范建立基于大语言模型的 SPEC 驱动开发流程，确保 SPEC 的编制、评审、迭代与代码生成活动有机衔接。

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。