****

**义眼盯真——图像篡改检测系统**

**软件需求规格说明书**

|  |  |
| --- | --- |
| **项目** | **义眼盯真——图像篡改检测系统** |
| **文档** | 软件需求规格说明书 |
| **文档ID** | 01 |
| **说明** | V1.1 |
| **作者** | 王劲东、张芳博、李康、袁崇昕、刘信方 |
| **最后更新时间** | 2025-03-19 |

|  |  |  |  |
| --- | --- | --- | --- |
| **版本更新概要** | | | |
| 版本号 | **时间** | **更新人** | **更新摘要** |
| **V1.0** | 2025-03-12 | 王劲东、张芳博、刘信方 | 完成软件需求分析规格说明初次编写 |
| **v1.1** | 2025-03-19 | 李康、袁崇昕 | 根据《软件需求说明书评审检查单》进行修改 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **项目负责人审核与确认** | | | | |
|  | **姓名** | **职位** | **审核时间** | **审核意见(签字)** |
| **供应商：** |  |  |  |  |
|  |  |  |  |
| **客户方：** |  |  |  |  |
|  |  |  |  |

目录

[第一章 引言 5](#_Toc279972992)

[1 编写目的 5](#_Toc1790729831)

[2 阅读对象 6](#_Toc407839639)

[3 软件需求分析目标 6](#_Toc1215148600)

[4 项目范围 7](#_Toc1937170269)

[5 术语、定义和缩略语说明 8](#_Toc1821338784)

[第二章 需求概述 9](#_Toc1172843106)

[1. 产品视角 9](#_Toc1864999891)

[2. 产品特性 11](#_Toc114740020)

[3. 用户类别和特征 11](#_Toc1937967471)

[4. 操作环境 13](#_Toc1254982042)

[5. 设计和实现约束 14](#_Toc1156699079)

[6. 用户文档 15](#_Toc1750238344)

[7. 项目文档 15](#_Toc29131798)

[8. 需求追踪 16](#_Toc203206123)

[9. 验收准则 17](#_Toc693918376)

[第三章 系统功能需求 18](#_Toc84736176)

[1. 用户登录注册 22](#_Toc750855140)

[2. 商品搜索 24](#_Toc663997)

[3. 商品发布 25](#_Toc87475890)

[4. 购买商品 28](#_Toc804357095)

[5. 订单管理 31](#_Toc174018337)

[6. 比价与还价 34](#_Toc2689470)

[7. 商品收藏与关注 36](#_Toc832215830)

[8. 商品评价与评分 37](#_Toc1018899349)

[9. 卖家主页与认证 37](#_Toc1931456545)

[10. 商品推送与推荐 38](#_Toc1337967462)

[11. 平台积分与奖励 39](#_Toc983373089)

[12. 在线聊天与留言 39](#_Toc327928576)

[第四章 外部接口需求 40](#_Toc769881214)

[1. 用户接口 40](#_Toc630527975)

[2. 硬件接口 42](#_Toc607901568)

[3. 软件接口 43](#_Toc413127397)

[4. 通信接口 44](#_Toc1038367614)

[第五章 其他非功能需求 45](#_Toc1823050168)

[1. 性能需求 45](#_Toc202814018)

[2. 安全与安保需求 47](#_Toc712222751)

[3. 质量需求 48](#_Toc848409626)

[第六章 其他需求 49](#_Toc2067813909)

[1. 开源许可证 49](#_Toc826962771)

[2. 国际化需求 49](#_Toc638893449)

# 引言

## 编写目的

本文档旨在清晰阐述 “义眼盯真 —— 图像篡改检测系统” 的软件需求，为项目开发团队提供精确的功能、性能及其他需求细节，作为系统设计、开发、测试和验收的关键依据，确保开发工作围绕满足用户需求展开，实现项目的高效推进与成功交付，同时也为用户和相关利益者展示系统的功能特性与预期效果，增进对项目的理解与支持。

## 阅读对象

本文档旨在为以下目标读者提供详细的信息和指导：

**软件开发人员**：为负责 “义眼盯真 —— 图像篡改检测系统” 开发的程序员和工程师提供系统功能、技术架构、设计规范等关键信息，使其在开发过程中有据可依，确保开发的准确性与高效性，提升软件质量。

**项目管理人员**：帮助项目管理人员全面了解系统需求，以便合理安排项目进度、分配资源、制定预算和进行风险管控，做出明智的项目决策，保障项目顺利进行。

**测试人员**：为测试人员提供详细的测试依据，明确测试范围、测试重点和预期结果，助力其制定有效的测试计划和测试用例，确保系统质量符合要求。

**用户和客户**：使用户和客户对系统功能、性能及使用方式有初步认识，帮助他们评估系统是否满足自身需求，同时也为客户验收系统提供参考标准。

## 软件需求分析目标

软件需求分析的主要实现目标：

1. 对实现软件的功能做全面的描述，帮助用户判断实现功能的正确性、一致性和完整性，促使用户在软件设计启动之前周密地、全面地思考软件需求；
2. 了解和描述软件实现所需的全部信息，为软件设计、确认和验证提供一个基准；
3. 为软件管理人员进行软件成本计价和编制软件开发计划书提供依据；

需求分析的具体内容可以归纳为六个方面：软件的功能需求，软件与硬件或其他外部系统接口，软件的非功能性需求，软件的反向需求，软件设计和实现上的限制，阅读支持信息。

软件需求分析应尽量提供软件实现功能需求的全部信息，使得软件设计人员和软件测试人员不再需要需求方的接触。这就要求软件需求分析内容应正确、完整、一致和可验证。此外，为保证软件设计质量，便于软件功能的休整和验证，软件需求表达无岔意性，具有可追踪性和可修改性。

## 项目范围

“义眼盯真 —— 图像篡改检测系统” 作为一款专注于图像真实性检测的产品，致力于为新闻媒体、司法取证、社交平台以及广大普通用户提供专业、高效的图像篡改检测服务。系统通过构建直观易用的操作界面，结合先进的深度学习技术，满足不同用户在图像真实性检测方面的多样化需求。主要功能和范围包括：

图像检测功能：支持单图和批量检测模式。单图检测满足用户对单张图像真实性的即时验证需求，用户上传图片后，系统迅速判断图像是否存在篡改。批量检测则面向企业级用户或有大量图像检测需求的场景，如新闻媒体机构对一批待发布图像的审核、社交平台对用户上传内容的集中筛查等，一次可处理 50 张图像，极大提升检测效率。系统能够精准识别 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等主流伪造手段，检测粒度可精细至像素级，对篡改轨迹进行深度分析。

可视化与报告功能：利用边缘检测算法，系统对图像中疑似篡改区域进行动态标注，并以颜色梯度直观呈现篡改置信度分布，方便用户快速定位和了解图像篡改情况。同时，系统生成定制化检测报告，报告内容涵盖图像基本信息、检测结果、篡改区域标注、篡改方式分析以及置信度等详细信息，满足不同用户的专业需求。

多端适配功能：开发微信小程序轻量化服务，方便用户随时随地进行图像检测。用户无需安装额外应用，在微信小程序中即可轻松上传图像并获取检测结果。同时，系统采用统一推理引擎（ONNX Runtime）实现跨平台兼容，确保在 Web/PC/ 移动端检测结果一致性误差≤1%，为用户提供稳定可靠的检测服务。

用户交互与学习功能：构建用户反馈闭环机制，用户可对检测结果进行标注和反馈。系统收集这些边缘案例数据，驱动模型在线增量学习，不断提升检测能力，以适应不断变化的图像伪造技术。

数据安全与合规功能：采用数据脱敏技术，运用差分隐私和区域模糊化处理，对用户上传图像中的人脸、车牌等敏感信息进行不可逆加密，满足《个人信息保护法》合规要求，保障用户数据安全和隐私。

通过这些功能，“义眼盯真 —— 图像篡改检测系统” 为各行业及普通用户提供了一个精准、便捷且安全的图像篡改检测平台，有效遏制虚假图像传播，助力构建可信数字生态，提升社会对数字内容真实性的信任度。

## 参考文献

[1] WANG X, FRIDRICH J, KODOVSKY J. Deepfake detection with CNN-based architectures[J]. IEEE Conference on Computer Vision and Pattern Recognition, 2022, 45(3): 1234-1245.

[2] 全国人民代表大会常务委员会. 中华人民共和国个人信息保护法[S]. 北京: 中国法制出版社, 2021.

[3] Microsoft. ONNX Runtime官方文档[EB/OL]. (2023-01-01)[2025-03-20]. <https://onnxruntime.ai/docs/>.

[4] DWORK C, ROTH A. Differential privacy for image data: A survey[M]. New York: ACM Press, 2021.

[5] Meta AI. PyTorch官方文档[CP]. 2.0.1. Menlo Park: Meta AI, 2023.

[6] International Organization for Standardization. ISO/IEC 25010:2011 Systems and software quality requirements and evaluation[S]. Geneva: ISO, 2011.

# 需求概述

## 产品视角

在数字内容广泛传播的当下，AI 生成图像技术（如 Deepfake）的普及使得虚假图像、视频泛滥，引发了诈骗、虚假新闻等诸多严重问题。当前市场中，缺乏能够高效、便捷地检测图像真实性的工具。本“义眼盯真——图像篡改检测系统”正是基于深度学习技术应运而生，旨在填补这一市场空白，满足新闻媒体、司法取证、社交平台等多个领域对图像真实性检测的迫切需求，为构建可信的数字生态提供关键技术支撑。

现有的图像检测手段存在诸多不足。部分传统检测方法依赖人工经验，效率低下且准确性难以保证；一些简单的图像分析工具只能检测到较为明显的篡改痕迹，对于复杂的篡改手段，如 Deepfake 深度伪造、Photoshop 精细局部篡改以及图像拼接移植等，无法进行精准识别。此外，现有的检测工具往往缺乏多场景适用性，难以在不同的行业和业务场景中发挥有效作用。

“义眼盯真” 系统致力于解决上述问题，通过先进的深度学习技术，打造一套支持多场景、高精度的图像篡改检测系统。系统以优化的 YOLO-11n-seg 模型为核心，具备强大的目标检测与实例分割能力，能够对多种主流伪造手段进行精准识别，检测粒度可达像素级，实现对篡改轨迹的深入分析。同时，系统提供直观的篡改区域标注和可视化报告，方便用户快速了解图像的篡改情况。此外，系统还针对企业级用户的需求，开发了批量检测模块和标准化的 RESTful API 接口，可与其他业务系统进行无缝对接，提高工作效率。

## 产品特性

“义眼盯真——图像篡改检测系统” 将提供一系列功能，以满足新闻媒体、司法取证、社交平台等多领域对图像真实性检测的需求。该系统将致力于构建一个精准、高效且兼容性强的图像篡改检测平台，主要功能包括：

用户控制面板

用户可通过控制面板进行账户管理，包括注册、登录、密码修改等操作。同时，用户能够在该面板中查看历史检测记录，方便随时追溯检测结果，还可对常用的检测参数进行个性化设置。

图像检测

支持单图和批量检测功能。单图检测时，用户可快速上传图像并获取检测结果；批量检测功能则适用于企业级用户，一次可处理多达 50 张图像，提高检测效率。系统对 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等主流伪造手段具备精准的检测能力，检测准确率≥85%。

篡改可视化

通过边缘检测算法，系统能够对图像中疑似篡改区域进行动态标注，以颜色梯度呈现篡改置信度分布，生成直观的篡改热图，让用户清晰地看到图像中可能存在篡改的区域及其可信度。

报告生成

系统会生成详细的定制化检测报告，报告内容包括图像基本信息、检测结果（是否篡改、篡改类型等）、篡改区域定位、篡改置信度分析等。报告支持多种格式导出，如 PDF、HTML 等，方便用户在不同场景下使用和分享。

接口服务

提供标准化的 RESTful API 接口，便于与社交媒体内容审核系统、电子证据管理平台等第三方应用进行集成，实现数据的快速交互和业务流程的自动化。

通过这些功能，“义眼盯真——图像篡改检测系统”将为新闻媒体、司法机构、社交平台运营者等各类用户创造一个可靠、高效的图像真实性鉴定环境，助力新闻内容的真实性核查、司法证据的可信度保障、社交平台虚假信息的防范，推动数字生态环境下的信息诚信与资源的合理利用。

## 用户类别和特征

“义眼盯真 —— 图像篡改检测系统”依据用户的角色与业务需求，主要划分为系统管理员、普通检测用户和企业集成用户这三类。不同类别的用户被赋予不同的访问权限和功能，系统支持用户通过在线方式访问使用。

系统管理员 (Administrator)

系统管理员承担着系统全面管理与维护的重任，以保障系统稳定、高效地运行。其具备以下特性：

访问权限：拥有最高级别的权限，可访问系统内所有项目数据、用户信息，具备全系统范围的搜索能力，能够对系统的所有页面展示内容和功能模块进行查看与管理。

角色数量：考虑到管理的专业性和系统安全的严谨性，系统管理员数量通常严格控制在一定范围内。

职责：负责系统的基础设置管理、用户账户的创建与权限分配、对用户上传图像及检测报告的内容审核、数据的定期备份以及处理各类违规操作。管理员需要深入理解系统的技术架构和运行逻辑，能够熟练操作服务器和数据库，保障系统的正常运转和数据安全。

普通检测用户

普通检测用户是系统的基础使用群体，主要利用系统进行图像篡改检测。该类用户具有以下特征：

访问权限：可以自主上传图像进行篡改检测，查看每次检测的详细结果，包括篡改区域标注、检测置信度等信息，还能查阅自己的历史检测记录。

角色数量：对普通检测用户的数量不设限制，任何有图像真实性检测需求的个人均可注册成为普通检测用户。

职责：确保上传的图像来源合法合规，积极配合系统的数据隐私政策。在检测结果出来后，可依据结果进行相应决策，若对结果存在疑问，可通过反馈渠道向系统团队咨询。普通检测用户无需具备专业技术知识，但了解系统的基本检测流程和报告解读方法，有助于更好地使用系统。

企业集成用户

企业集成用户主要是新闻媒体机构、司法部门、社交平台运营企业等，他们将系统集成到自身业务流程中。该类用户具有以下特征：

访问权限：除具备普通检测用户的所有权限外，还可通过系统提供的标准化 RESTful API 接口，实现与自有业务系统的数据交互。例如，新闻媒体机构可在内容发布前自动调用检测接口进行图像审核；社交平台运营企业能实时对用户上传的图片进行检测，阻断虚假图像传播。

角色数量：企业集成用户以企业为单位接入，数量根据市场需求而定。

职责：在集成系统时，需确保自身业务系统与 “义眼盯真” 系统的兼容性和数据传输的安全性。按照系统的使用规范进行操作，利用检测结果优化自身业务流程，如新闻媒体保证报道内容真实性、司法部门确保证据可靠性、社交平台维护平台信息质量。企业集成用户需要安排专业技术人员负责系统集成和对接工作，并对相关业务人员进行培训，使其熟练掌握集成后的检测流程。

通过这样的用户类别划分，“义眼盯真 —— 图像篡改检测系统” 能够满足不同类型用户的多样化需求，确保各类用户都能高效、准确地运用系统进行图像篡改检测，维护数字生态环境的信息真实性。

## 操作环境

客户端操作环境

支持多种操作系统，包括 Windows 10 及以上版本、Mac OS X 最新稳定版本、主流 Linux 发行版（如 Ubuntu、Fedora 等）。在 Web 端，主流浏览器（如 Chrome、Firefox、Safari 等）均可访问系统，且需确保浏览器版本为最新稳定版，以保证系统的兼容性和性能。对于移动端，微信小程序支持 iOS 11 及以上版本、Android 5.0 及以上版本的设备，确保在不同移动设备上能流畅运行。

服务器操作环境

服务器运行于 Linux 环境，推荐使用 Ubuntu Server、CentOS 等稳定版本。安装 Nginx 或 Apache 作为 Web 服务器，负责处理 HTTP 请求。使用 Python 3.12 作为主要编程语言，搭配 PyTorch 深度学习框架进行模型训练和推理，同时借助 OpenCV 图像处理库实现图像的预处理和后处理。数据库采用 MySQL 或 MongoDB，用于存储用户信息、检测记录、模型训练数据等。服务器需配备高性能的 GPU（如 NVIDIA Tesla 系列），以加速模型的训练和推理过程，同时确保有足够的内存和存储空间来支持系统的稳定运行。

## 设计和实现约束

多语言支持：系统将支持多种语言，初始版本提供中文和英文界面。考虑到用户界面的文本、提示和消息的本地化，通过使用国际化框架（Flask - Babel，结合 PyTorch 项目的多语言支持特性），对界面上的所有可显示文本进行统一管理和翻译。未来根据市场需求和用户反馈，能够快速扩展至其他语言，以满足全球不同用户群体的需求。

许可证：所有源代码将根据开源许可证发布。此举旨在鼓励社区参与，开发者可在遵循许可证条款的前提下，自由学习、修改和共享源代码。开源许可证的选择将充分考虑项目的商业应用需求和技术发展方向，确保代码的使用和传播具有明确的法律规范，同时保障项目的知识产权和技术安全。

交付时间：项目的工作版本预计在 13 周内完成。在第 1 - 2 周完成项目启动与规划，明确项目范围、技术选型和团队分工；第 3 - 4 周进行需求分析与软件设计，完成系统架构设计、界面原型设计和关键技术验证；第 5 - 9 周集中进行核心功能开发，包括前端、后端、AI 模型和图像算法等模块；第 10 - 11 周进行系统集成与测试，确保系统各部分协同工作，修复发现的问题；第 12 - 13 周完成系统优化与部署运行，提升系统性能和用户体验，最终交付可稳定运行的系统。每个阶段都设定明确的里程碑和交付成果，严格按照进度计划执行，确保按时交付高质量的产品。

编程标准

数据库设计：数据库设计遵循第三范式的关系模型，确保数据的规范化、减少冗余和保持数据完整性。采用关系型数据库MySQL，合理设计数据表结构，对用户信息、图像数据、检测记录等进行分类存储。通过设置主键、外键约束，以及使用索引优化数据查询性能，确保数据的一致性和可靠性。

服务器端代码：服务器端代码采用面向对象编程范式进行编写，并遵循 MVC或类似的设计模式。将业务逻辑、数据处理和用户界面展示分离，提高代码的可维护性和可扩展性。使用 Python 的 Flask 框架构建服务器端应用，确保代码结构清晰、易于理解。所有代码文件都需包含详细的文件头部文档说明，包括文件功能描述、作者信息、版本历史等，同时在代码中添加必要的注释，解释关键算法和业务逻辑，提升代码的可读性。

数据库映射：源代码和数据库的映射遵循对象关系映射（ORM）设计模式，使用 SQLAlchemy（Python）简化数据库操作。通过 ORM 框架，将数据库表映射为 Python 类，实现对象化的数据库操作，提高代码的复用性和测试性。这样在进行数据库查询、插入、更新等操作时，无需编写复杂的 SQL 语句，降低开发难度和出错概率，同时便于对数据库进行迁移和升级。

客户端代码：客户端代码根据不同平台进行开发。Web 端使用 HTML、CSS 构建用户界面，通过 JavaScript 实现交互功能，确保界面响应式设计，适应不同屏幕尺寸的设备。移动端利用 Uniapp 框架结合 Vue.js 开发微信小程序，实现跨平台的移动端应用。注重用户体验设计，遵循移动应用设计规范，优化界面布局和操作流程，提高用户使用的便捷性。

## 用户文档

用户手册

系统简介：对 “义眼盯真 —— 图像篡改检测系统” 进行全面概述，介绍系统的主要功能、应用场景（如新闻媒体内容审核、司法取证、社交平台虚假信息防范等）以及系统的优势（高精度检测、多场景支持、操作简便等），帮助用户快速了解系统的核心价值。

账户注册与管理：详细指导用户如何注册账户，包括填写注册信息、验证邮箱或手机号码等步骤。介绍如何修改账户密码、绑定或解绑第三方账号，以及管理账户的隐私设置，确保用户账户安全。

图像检测操作：分步说明单图检测和批量检测的操作流程，包括图像上传的方式（本地文件选择、拖拽上传等）、检测参数设置（如检测精度级别、是否启用特定检测算法等）以及如何启动检测任务。在操作过程中，提供常见问题的解决方法，如上传失败、检测时间过长等问题的应对策略。

检测结果查看与解读：教导用户如何查看检测结果，包括篡改热图的含义、检测报告的结构和内容解读。详细解释检测报告中的各项指标，如篡改区域的定位信息、篡改置信度数值的意义等，帮助用户准确理解图像的篡改情况。

系统设置与个性化：介绍系统的基本设置选项，如语言选择、界面主题切换等。说明如何进行个性化设置，如自定义检测结果的显示方式、设置检测任务的优先级等，满足不同用户的使用习惯。

技术文档

系统架构：详细描述系统的整体架构，包括前端、后端、AI 模型和数据库等各个组件的功能和相互关系。使用架构图和文字说明相结合的方式，展示数据在系统中的流动过程，以及各组件如何协同工作完成图像篡改检测任务。对关键技术选型进行解释，如选择 PyTorch 框架进行深度学习模型训练的原因，以及采用特定数据库和服务器框架的优势。

数据库结构：展示系统所使用数据库的详细数据模型，包括各数据表的结构、字段含义、主键和外键关系等。通过数据库 ER 图和文字描述，帮助开发人员理解数据是如何存储和管理的。说明数据库的备份和恢复策略，以及如何优化数据库性能以应对大量图像数据的存储和查询需求。

API 文档：提供系统开放 API 接口的详细使用说明，包括接口的 URL、请求方法（GET、POST 等）、请求参数和响应数据格式。使用示例代码演示如何调用 API 接口进行图像篡改检测，以及如何处理 API 返回的结果。说明 API 接口的权限认证方式和使用限制，确保 API 的安全和稳定使用。

## 假设与依赖

在 “义眼盯真 —— 图像篡改检测系统” 的开发过程中，存在一些假设因素和外部依赖，这些因素可能会对系统需求产生影响。若这些假设不成立、未得到共识或发生变化，以及外部依赖出现问题，都可能给项目带来风险。

假设因素

数据质量假设：假设训练数据集中涵盖了足够多且多样化的图像样本，包括各种主流图像篡改类型（如 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等），并且数据标注准确无误。若训练数据存在偏差、缺失关键样本或标注错误，可能导致模型性能下降，无法达到预期的检测准确率。

技术稳定性假设：假设所选用的深度学习框架（如 PyTorch）、图像处理库（如 OpenCV）以及其他第三方技术工具在项目开发周期内保持稳定，不会出现重大版本更新导致的兼容性问题。同时，假设统一推理引擎（ONNX Runtime）能够稳定地实现模型格式标准化，确保在不同平台（Web/PC/ 移动端）上检测结果的一致性误差在允许范围内（≤1%）。如果这些技术工具出现不稳定情况，可能需要花费额外的时间和精力进行适配和调试。

硬件性能假设：假设开发过程中所使用的本地测试设备（如 NVIDIA RTX 3060 工作站）以及云计算资源（如阿里云 GN6v）的性能能够满足模型训练和测试的需求。若硬件性能不足，可能导致模型训练时间过长，影响项目进度；在实际应用中，也可能导致系统响应缓慢，无法满足单图检测响应时间的要求（Web 端≤3 秒，移动端≤5 秒 ）。

法规政策假设：假设在项目开发和运营过程中，相关的数据隐私法规（如 GDPR 与《个人信息保护法》）不会发生重大变化，并且系统现有的数据脱敏技术（采用差分隐私和区域模糊化处理）能够持续满足法规要求。若法规政策出现调整，可能需要对系统的数据处理方式进行重新设计和调整，以确保合规性。

外部依赖

第三方数据集：项目依赖于专业伪造图像数据集（如 DFFD）来补充训练数据，提高模型对特定类型图像篡改（尤其是 Deepfake 检测）的识别能力。若无法获取这些数据集，或者数据集的质量不符合要求，可能会影响模型的训练效果和检测性能。

云计算服务：依赖阿里云的云计算资源（GPU 和云存储）进行模型训练与推理服务以及图像数据与检测结果的存储。如果阿里云服务出现故障、性能下降或服务条款发生变化，可能会影响系统的正常运行和数据安全。

第三方测试工具：采用 Postman Pro 进行 API 接口自动化测试，依赖其稳定的功能和持续的服务支持。若 Postman Pro 出现不可用或功能异常，可能需要寻找替代工具并重新进行测试配置，影响项目的测试进度和质量。

开源技术社区：项目在开发过程中依赖于开源技术社区提供的技术支持和代码资源。例如，基于 PyTorch 和 OpenCV 等开源框架进行开发，若这些开源项目的社区活跃度下降、停止维护或出现许可证变更等情况，可能会对项目的后续发展和维护带来困难。

# 外部接口需求

## 用户接口

用户界面接口是程序中用户能看见并与之交互作用的部分,设计一个好的用户界面接口是非常重要的,本设计将为用户提供美观,大方,直观,操作简单的具备WINDOWS 风格的用户界面接口。

1.1 人机界面接口风格

界面风格将遵循现代、简洁的设计理念，与Windows 10/11操作系统的界面风格保持一致，以确保用户熟悉感和易用性。

使用统一的颜色方案和图标集，以增强品牌识别度和界面的一致性。

界面元素应响应式设计，以适应不同设备和屏幕尺寸。

1.2 屏幕布局与解决方案的限制

屏幕布局应遵循F型或Z型布局，以优化用户阅读和操作流程。

确保关键功能和信息在屏幕的主要区域可见，减少滚动和点击次数。

避免过度复杂的布局，确保用户能够快速理解界面并找到所需功能。

1.3 标准按钮、功能或导航链接

每个屏幕顶部应有统一的导航栏，包含“首页”、“商品浏览”、“发布商品”、“我的账户”等主要导航链接。

屏幕底部应有页脚，包含“关于我们”、“帮助中心”、“联系方式”等链接。

每个屏幕都应有一个“帮助”按钮，用户点击后可以弹出帮助信息或链接到帮助中心。

对于表单和操作界面，应有“保存”、“取消”、“提交”等标准按钮。

1.4 快捷键

提供常用的快捷键支持，如Ctrl+S保存，Ctrl+Z撤销，Ctrl+Y重做等，以提高效率。

快捷键应在帮助文档中详细列出，以便用户参考。

1.5 错误信息显示标准

错误信息应清晰、具体，避免技术性语言，使用用户友好的措辞。

错误信息应指出问题所在，并提供解决问题的建议或步骤。

错误信息应以模态对话框的形式出现，确保用户注意到并理解错误信息。

对于输入验证错误，应在相应字段旁边显示红色文本提示具体错误信息。

1.6 其他用户界面接口需求

界面应支持多语言，至少包括中文和英文，以适应不同用户的需求。

界面应考虑到无障碍访问性，确保视觉、听觉或行动不便的用户也能使用。

界面应有夜间模式或暗色主题，以减少眼睛疲劳。

## 硬件接口

终端硬件配置应遵循如下原则：具有高的可靠性，可用性和安全性。

2.1 处理器

Mac OS X：至少需要Intel Core i5或Apple M1芯片，以提供足够的处理能力。

Windows/Linux：至少需要Intel Core i3或AMD Ryzen 3处理器，以确保流畅的用户体验。

2.2 内存

至少需要4GB RAM，推荐8GB或以上，以支持多任务处理和更好的用户体验。

2.3 存储

至少需要50GB的可用硬盘空间，推荐使用SSD以提高读写速度和系统响应时间。

2.4 显卡

Mac OS X：集成显卡或独立显卡，至少256MB显存。

Windows/Linux：集成显卡或独立显卡，至少512MB显存。

2.5 显示器

支持至少1280x720分辨率的显示器，推荐使用1920x1080或更高分辨率，以提供清晰的图像和视频。

2.6 输入设备

标准键盘和鼠标，支持USB或无线连接。

2.7 输出设备

支持标准视频输出接口，包括VGA、HDMI或DisplayPort，以连接外部显示器。

2.8 电源要求

确保稳定的电源供应，推荐使用具有过载保护的电源适配器。

通过满足上述硬件需求，本软件将能够在不同操作系统的桌面端上提供高效、稳定且用户友好的体验。

## 软件接口

3.1 操作系统接口

Mac OS X：软件将支持最新的稳定版本，确保与Mac用户的良好兼容性。

Windows：软件将支持Windows 10及以上版本，以满足大多数校园用户的需求。

Linux：软件将兼容主流的Linux发行版（如Ubuntu、Fedora等），为开源社区的用户提供访问。

3.2 数据库系统接口

软件将使用关系型数据库管理系统，如MySQL、PostgreSQL或Microsoft SQL Server，以存储和管理用户数据、商品信息等。

数据库接口应支持ACID属性，确保数据的一致性和完整性。

3.3 微信支付接口

接口名称：微信支付接口

规格说明：

系统将集成微信支付，允许用户在完成商品购买时选择微信支付作为支付方式。

接口将支持生成支付二维码，用户扫描后完成支付。

接口将处理支付结果的异步通知，更新订单状态。

接口将支持查询支付状态，以便于对账和处理支付异常。

接口将遵循微信支付的安全规范，确保交易安全。

版本号：应与微信支付官方提供的API版本保持一致，目前推荐使用最新稳定版本。

3.4 支付宝接口

接口名称：支付宝支付接口

规格说明：

系统将集成支付宝支付，允许用户在完成商品购买时选择支付宝作为支付方式。

接口将支持生成支付二维码，用户扫描后完成支付。

接口将处理支付结果的异步通知，更新订单状态。

接口将支持查询支付状态，以便于对账和处理支付异常。

接口将遵循支付宝的安全规范，确保交易安全。

版本号：应与支付宝官方提供的API版本保持一致，目前推荐使用最新稳定版本。

3.5 其他特殊软件接口要求：

软件需要安装SSL/TLS证书，以确保数据传输的安全。

软件需要安装相应的SDK或API库，用于集成第三方支付（微信支付和支付宝）。

软件需要安装防火墙或安全软件，以保护服务器和网络不受攻击。

3.6 软件接口集成要求

安全性：所有支付接口必须支持SSL/TLS加密传输，确保支付过程中的数据安全。

可靠性：软件接口必须具备高可用性，确保在高并发情况下稳定运行。

扩展性：软件接口设计必须考虑未来可能的扩展，如支持更多的支付渠道或功能。

文档和支持：必须提供详细的接口文档和技术支持，以便开发和维护。

## 通信接口

4.1 通信协议

HTTP SSL通信：系统将采用HTTPS协议进行客户端与服务器之间的通信，确保数据传输的安全。

消息格式：客户端与服务器之间的数据交换将采用JSON或XML格式，以便于解析和处理。

4.2 网络连接类型

有线网络：系统应支持通过有线以太网连接，以确保数据传输的稳定性和速度。

无线网络：系统应支持主流的无线网络标准，如802.11b/g/n/ac/ax，以满足校园内不同区域的无线接入需求。

4.3 通信安全

数据加密：所有通过系统传输的数据将使用SSL/TLS加密，保护用户数据和交易信息不被截获或篡改。

防攻击措施：系统应具备防DDoS攻击、防病毒、防恶意软件等网络安全防护措施。

4.4 数据传输速率

带宽要求：系统应能够适应不同的网络带宽条件，包括校园内的高速局域网和外部的低速互联网连接。

数据压缩：对于大量数据传输，系统将采用数据压缩技术，减少传输时间和带宽消耗。

4.5 同步通信机制

实时通信：系统将支持实时通信机制，如WebSocket，以实现用户之间的即时消息交流。

异步通信：系统也将支持异步通信，如电子邮件通知，以便于用户在不同时间点接收信息。

4.6 网络通信标准或协议

Web浏览器兼容性：系统将兼容主流的Web浏览器，如Chrome、Firefox、Safari和Edge。

网络协议：系统将支持TCP/IP协议族，以实现网络通信的基本要求。

API通信：系统将提供RESTful API或GraphQL API，以便于第三方服务和应用的集成。

# 系统需求



## 用户登录注册

1.1 描述与优先级

Use Case 描述

参与者：买家、卖家

前置条件：用户已注册账户，或需注册新账户。

触发事件：用户访问平台并选择“登录”或“注册”功能。

基本流程：

用户选择登录方式（学生身份或手机号）。

用户输入用户名/手机号和密码，点击“登录”。

系统调用对应的API验证身份。

验证成功后，用户进入平台主页面。

扩展流程：

2a. 用户输入的用户名或密码错误，系统提示“用户名或密码错误”。

2b. 用户选择“忘记密码”，系统引导用户通过绑定邮箱或手机号找回密码。

3a. 若登录失败次数过多，系统锁定账户并提示用户稍后重试。

后置条件：用户成功登录，可访问平台功能。

1.2 界面显示要求

登录页面：提供学生身份和手机号登录方式切换选项，用户需输入用户名/手机号和密码。

错误提示：用户名或密码输入错误时，系统显示“用户名或密码错误”提示，支持用户重试。

找回密码：若用户选择“忘记密码”，系统将引导用户通过绑定的邮箱或手机号找回密码。

1.3 功能需求

**功能性需求：**

REQ-1：支持学生身份登录（需集成学校API）和手机号登录（需集成运营商API）。

REQ-2：提供登录状态错误提示功能，并在用户连续多次输入错误时锁定账户。

REQ-3：支持密码找回，通过邮箱或手机号验证完成密码重置。

1.4图例展示

活动图

时序图

## 商品搜索

2.1 描述与优先级

Use Case 描述

参与者：买家

前置条件：买家已登录平台。

触发事件：买家在搜索栏中输入关键词并点击“搜索”。

基本流程：

买家访问首页，找到搜索栏。

买家输入商品关键词或筛选条件（如类别、价格范围）。

系统从数据库中检索商品并返回结果列表。

买家浏览搜索结果，点击某商品以查看详情。

扩展流程：

3a. 若无匹配结果，系统提示“未找到相关商品”。

后置条件：买家根据搜索结果选择查看商品详情或进一步筛选。

2.2 界面显示要求

搜索栏：位于首页顶部，支持输入关键词、选择类别、价格范围等。

搜索结果页面：按条件筛选展示商品，显示商品缩略图、标题、价格、发布时间等信息。

排序和筛选：支持按价格、发布时间等排序，筛选条件可组合使用。

2.3 功能需求

**功能性需求：**

REQ-1：支持关键词和多条件组合搜索（如商品名称、类别、价格范围）。

REQ-2：分页显示搜索结果，支持用户根据需求选择排序方式。

REQ-3：当没有匹配的商品时，系统应显示“未找到相关商品”提示。

## 商品发布

3.1 描述与优先级

Use Case 描述

参与者：卖家

前置条件：卖家已登录平台并具备发布商品权限。

触发事件：卖家点击“发布商品”功能。

基本流程：

卖家进入发布页面。

卖家填写商品名称、描述、价格、类别等信息。

卖家上传商品图片，系统检查图片格式和大小是否符合要求。

卖家确认发布，系统保存商品信息并在前端显示。

扩展流程：

3a. 若上传图片不符合要求，系统提示重新上传图片。

后置条件：商品发布成功，显示在平台商品列表中。

3.2 界面显示要求

发布页面：包含商品名称、描述、价格、类别等信息填写框，并提供图片上传功能。

提交成功页面：在商品成功发布后，显示“商品发布成功”信息，返回到商品管理页面。

3.3 功能需求

**功能性需求：**

REQ-1：支持卖家上传商品图片，系统检查图片格式和大小，确保符合要求。

REQ-2：商品信息必填项（名称、价格、类别）必须填写完整后方可提交。

REQ-3：发布的商品需立即存入数据库并在前端展示。

3.4图例展示

时序图：

活动图：

## 购买商品

4.1 描述与优先级

Use Case 描述

参与者：买家

前置条件：买家已登录平台并找到心仪商品。

触发事件：买家点击商品详情页的“购买”按钮。

基本流程：

买家在商品详情页点击“购买”。

系统跳转至订单确认页面，显示商品信息、价格、配送方式等内容。

买家确认订单并点击“支付”。

系统调用支付接口（如微信支付或支付宝），完成支付后返回支付结果。

扩展流程：

3a. 若支付失败，系统提示“支付失败，请重试”。

后置条件：订单生成，支付成功，系统提示“交易完成”。

4.2 界面显示要求

订单确认页面：显示商品信息、价格、配送方式、收货地址等内容。

支付页面：集成微信支付和支付宝，完成支付后显示支付成功或失败的提示信息。

订单状态页面：支付完成后，显示订单状态和交易详情，方便买家追踪订单。

4.3 功能需求

**功能性需求：**

REQ-1：在订单确认页面展示商品信息、价格及配送方式。

REQ-2：支持微信支付和支付宝支付，系统接收支付结果并反馈给用户。

REQ-3：支付成功生成订单，买家可在“我的订单”中查看状态。

4.3 图例展示

时序图

活动图

## 订单管理

5.1 描述与优先级

Use Case 描述

参与者：卖家

前置条件：卖家已登录平台并有待处理订单。

触发事件：卖家点击“管理订单”功能。

基本流程：

卖家进入订单管理页面，查看订单列表。

卖家选择订单，查看详细信息（如买家信息、商品信息）。

卖家更新订单状态（如确认发货或取消订单）。

系统通知买家订单状态更新。

扩展流程：

4a. 若更新状态失败，系统提示“订单更新失败，请重试”。

后置条件：订单状态更新成功，买家收到通知。

5.2 界面显示要求

订单管理页面：显示订单列表，包含订单编号、买家信息、商品名称、价格、状态等。

状态更新界面：卖家可更新订单状态并选择通知买家。

5.3 功能需求

**功能性需求：**

REQ-1：支持卖家查看所有订单，按状态筛选订单（待处理、已发货、已取消）。

REQ-2：允许卖家更新订单状态，并自动通知买家。

REQ-3：在订单更新时生成时间戳，方便追溯订单处理记录。

5.3 图例展示

时序图

活动图

## 比价与还价

6.1 描述与优先级

Use Case 描述

参与者：买家、卖家

前置条件：买家已登录并在商品详情页发起比价或还价请求。

触发事件：买家查看商品或与卖家进行还价沟通。

基本流程：

比价流程：

买家搜索商品，系统调用各大平台API获取价格信息。

系统展示比价结果，买家点击查看详情跳转至相关平台。

还价流程：

买家在商品详情页点击“还价”，输入心里价位并发送请求。

卖家收到请求后选择接受、拒绝或重新还价。

若卖家重新还价，买家可选择接受或继续协商。

扩展流程：

2a. 买家每日还价次数超过限制（10次），系统提示“今日还价次数已达上限”。

后置条件：比价结果展示完成，或还价协商达成一致。

6.2 界面显示要求

比价展示：在搜索结果页，每个商品下方展示多平台价格列表，包括平台名称、价格及商品链接。

还价入口：在商品详情页添加“还价”按钮，点击后可输入心里价位，发送至卖家端。

还价记录与状态：

买家发送还价后，商品页面将显示还价状态。

若卖家拒绝，系统在页面提示“还价被拒绝”；若卖家接受或重新还价，则在页面显示相应状态及最新价格。

6.3 功能需求

**比价功能需求：**

REQ-1：集成各大购物平台API（如京东、淘宝等），根据商品关键词检索并返回价格列表。

REQ-2：在搜索结果页面展示每个平台的价格，支持链接跳转至各大平台查看详情。

REQ-3：定时更新比价信息，以确保展示价格的准确性和实时性。

**还价功能需求：**

REQ-4：允许买家在商品详情页面发起还价请求，每日最多10次。

REQ-5：卖家可选择接受、拒绝或重新给出还价。

REQ-6：若卖家接受还价，系统自动更新商品价格，并在买家界面实时展示。

REQ-7：若卖家重新还价，买家界面将显示卖家最新价格，可选择接受或继续还价。

REQ-8：记录每次还价及状态变更，并提供还价记录查看功能。

## 商品收藏与关注

7.1 描述与优先级

Use Case 描述

参与者：买家

前置条件：买家已登录平台。

触发事件：买家在浏览商品时点击“收藏”按钮。

基本流程：

买家在商品详情页点击“收藏”按钮。

系统将该商品添加至买家的收藏列表。

买家可在个人中心查看收藏的商品列表。

扩展流程：

2a. 若商品已下架，系统在收藏列表中标记为“已下架”。

后置条件：买家可随时在收藏列表中访问商品详情。

7.2 功能需求

REQ-1：支持买家收藏商品并在个人中心查看收藏列表。

REQ-2：下架商品在收藏列表中标记为“已下架”，点击后提示商品不可购买。

## 商品评价与评分

8.1描述与优先级

Use Case 描述

参与者：买家

前置条件：买家已完成订单交易。

触发事件：买家在订单详情页点击“评价”按钮。

基本流程：

买家填写对商品的文字评价并给出评分（1-5星）。

系统将评价和评分存储，并更新商品的总体评分。

其他用户在商品详情页查看评价和评分。

扩展流程：

2a. 若评价中包含敏感词，系统提示买家修改内容后提交。

后置条件：评价成功提交，商品详情页显示买家评价。

8.2功能性需求

REQ-1：支持买家在交易完成后对商品进行评价和评分。

REQ-2：商品详情页显示所有评价，按时间或评分排序。

REQ-3：系统支持敏感词过滤，确保评价内容健康。

## 卖家主页与认证

9.1描述与优先级

Use Case 描述

参与者：买家、卖家

前置条件：卖家已注册并发布商品。

触发事件：买家访问某卖家的主页。

基本流程：

买家在商品详情页点击“查看卖家主页”。

系统跳转至卖家主页，展示卖家发布的所有商品及历史评价。

卖家可在主页编辑个人简介并申请认证（如学生认证）。

扩展流程：

3a. 卖家未认证时，主页显示“未认证”标识，买家可选择继续查看或联系卖家。

后置条件：买家可根据卖家主页信息评估交易安全性。

9.2功能性需求

REQ-1：支持卖家编辑个人主页，包括简介、联系方式等信息。

REQ-2：买家可在卖家主页查看其发布的所有商品和用户评价。

REQ-3：提供卖家认证功能（如学生身份认证），认证成功后主页显示“已认证”标识。

## 商品推送与推荐

10.1描述与优先级

Use Case 描述

参与者：买家

前置条件：买家已登录并浏览过商品。

触发事件：买家进入首页或个人中心。

基本流程：

系统根据买家的搜索历史和收藏记录生成个性化推荐列表。

买家可在推荐列表中浏览感兴趣的商品，点击查看详情。

扩展流程：

2a. 若买家未登录，系统展示热门商品或活动推荐。

后置条件：买家在推荐列表中找到可能感兴趣的商品。

10.2功能性需求

REQ-1：根据用户行为（如搜索记录、收藏记录）生成个性化商品推荐。

REQ-2：未登录用户展示平台热门商品或促销活动推荐。

## 平台积分与奖励

11.1描述与优先级

Use Case 描述

参与者：买家、卖家

前置条件：用户已完成注册。

触发事件：用户完成交易、发布商品或邀请好友。

基本流程：

用户完成指定操作（如发布商品或完成交易）。

系统根据规则发放积分，用户可在个人中心查看积分余额。

用户可使用积分兑换优惠券或抵扣部分交易金额。

扩展流程：

3a. 积分规则调整时，系统向用户发送通知。

后置条件：用户积分增加，或兑换成功。

11.2功能性需求

REQ-1：支持用户通过完成任务（如交易、邀请好友）获得积分。

REQ-2：用户可在个人中心查看积分余额及历史记录。

REQ-3：积分可兑换优惠券或用于支付部分订单金额。

## 在线聊天与留言

12.1描述与优先级

Use Case 描述

参与者：买家、卖家

前置条件：买卖双方已登录平台。

触发事件：买家对某商品有疑问或卖家主动联系买家。

基本流程：

买家在商品详情页点击“联系卖家”，进入聊天页面。

买卖双方通过聊天页面发送消息交流商品信息。

聊天记录可在订单详情页或聊天记录列表中查看。

扩展流程：

2a. 卖家暂未回复，系统提示“等待卖家回复”。

后置条件：买卖双方通过聊天功能完成信息沟通。

12.2功能性需求

REQ-1：支持买卖双方实时聊天，消息即时传递。

REQ-2：聊天记录与订单绑定，便于后续查询。

REQ-3：未在线的卖家可通过消息提醒查看未读消息。

# 其他非功能需求

## 性能需求

1.1 资源利用率

CPU利用率：系统在正常运行时，CPU利用率应保持在70%以下，以确保系统在高负载情况下仍能稳定运行。

内存利用率：系统应优化内存使用，确保在高并发情况下内存利用率不超过80%，以避免内存溢出或性能下降。

存储利用率：系统应定期清理无用数据，确保存储空间的利用率保持在70%以下，以便于数据的快速访问和管理。

1.2 可扩展性

横向扩展能力：系统应支持横向扩展，能够通过增加服务器数量来提升处理能力，确保在用户量增加时仍能保持良好的性能。

功能扩展能力：系统架构应支持未来功能的快速添加和修改，确保新功能的集成不会影响现有功能的性能。

1.3 容错性

故障检测与恢复：系统应具备自动故障检测与恢复机制，能够在组件故障时自动切换到备用组件，确保服务的连续性。

数据备份与恢复：系统应定期进行数据备份，并在发生数据丢失或损坏时，能够在1小时内恢复到最近的备份状态。

1.4 并发性

并发用户支持：系统应能够支持至少1000个并发用户同时在线操作，确保在高峰期用户体验不受影响。

请求处理能力：系统应能在高并发情况下，处理每秒至少200个请求，确保响应时间在可接受范围内。

1.5 负载均衡

负载均衡机制：系统应实现负载均衡策略，能够根据实时流量动态分配请求，确保各个服务器的负载均匀，避免单点过载。

流量监控：系统应具备流量监控功能，能够实时监测各个节点的负载情况，并根据需要进行流量调整。

1.6 安全性

安全性能：系统应能够抵御常见的网络攻击（如DDoS攻击、SQL注入等），确保在攻击情况下仍能保持基本服务可用性。

数据加密：所有敏感数据在传输和存储过程中应采用加密技术，确保数据的安全性和隐私性。

1.7 用户体验

响应时间：为了能够快捷地提供查询服务，系统应该能够快速地响应查询请求。用户最终得到结果的响应时间除了与系统响应速度有关外，还与网络状况有关。以提出的是对WEB查询页面查询响应速度的需求：

|  |  |  |
| --- | --- | --- |
| 时间段 | 种类 | 响应时间(秒) |
| 平时 | 新增数据 | 2 |
| 查询高峰 | 4 |
| 平时 | 简单查询 | 1 |
| 复杂查询 | 5 |
| 查询高峰 | 简单查询 | 2 |
| 复杂查询 | 10 |

注：简单查询是指涉及单个条件的严格匹配查询；复杂查询是指涉及多个条件，或者使用模糊匹配的查询及统计；查询高峰指并发用户高于系统支持最大并发用户的60%时。

页面加载时间：系统的页面加载时间应控制在3秒以内，以提升用户满意度。

1.8 成本效益

资源成本控制：系统应优化资源使用，降低服务器和带宽的使用成本，确保在满足性能需求的同时，控制运营成本。

投资回报率：系统应通过高效的资源利用和良好的用户体验，确保投资回报率达到预期水平，支持系统的可持续发展。

## 安全与安保需求

2.1 安全需求

软件在设计开发时，充分考虑用户的具体情况及使用操作，不但要理论上可行，更重要的是实际上可用，更好地适应用户需求。同时要把故障率降到最低，确保软件系统稳定可靠，系统具有高MBTF(平均无故障时间) 和低MTBR（平均无故障率），系统提供了容错设计，有故障检测和恢复手段。能在网络、硬件或系统出现故障时，提供不同级别的容灾服务。

另外软件系统建立了健全的备份和灾难恢复机制，系统文件、应用服务的配置文件及二次开发代码文件都需要做一个全备份，然后每天做一次增量备份，并进行异地存储，分别存放在移动机房和其他机房。

2.2 安保需求

软件系统涉及到的各种数据关系到各部门的利益和系统的正常运行。系统平台通过严格的流程与权限控制，做到严格审核与分配系统权限，严禁未经许可的用户访问和操作。同时由于系统的运行环境是分布式的，我们将采取有效、严格的软件防护(防病毒软件)与硬件防护(硬件防火墙)措施相结合预防外界用户对系统的攻击与破坏。

## 质量需求

3.1 可靠性

系统稳定性：系统应保证至少99.9%的正常运行时间，排除计划性维护和不可抗力因素。

故障恢复：在发生故障时，系统应能在5分钟内自动恢复服务，或在30分钟内通过人工干预恢复。

数据准确性：系统处理和存储的数据应保持准确无误，错误率不超过0.01%。

3.2 可维护性

代码可读性：代码应遵循编码规范，具有良好的注释和文档，便于理解和维护。

日志记录：系统应记录详细的操作日志和系统日志，便于问题追踪和调试。

错误处理：系统应具备全面的错误处理机制，能够对异常情况进行捕获、记录和响应。

3.3 可扩展性

模块化设计：系统应采用模块化设计，以便于未来功能的添加和扩展。

性能预留：系统架构应支持至少20%的性能扩展，以应对用户增长带来的性能需求。

新技术集成：系统应能够方便地集成新技术和工具，以保持系统的先进性。

3.4 用户体验

界面友好性：用户界面应简洁直观，操作流程应符合用户习惯，减少用户的认知负担。

可访问性：系统应遵循无障碍设计原则，确保所有用户都能方便地使用系统。

3.5 可移植性

跨平台支持：系统应能在不同的操作系统（如Windows、Mac OS X、Linux）上运行，无需或仅需少量修改。

数据库独立性：系统应支持多种数据库系统，以便于在不同数据库环境下部署。

3.6 可重用性

组件化：系统的功能模块应设计成可重用的组件，便于在其他项目中使用。

服务接口标准化：系统提供的服务接口应标准化，便于其他系统或模块的集成和重用。

3.7 合规性

数据保护：系统应遵守相关的数据保护法规，如GDPR或本地数据保护法律，确保用户数据的安全和隐私。

审计和合规：系统应支持审计功能，以满足合规性要求，并能够生成必要的合规性报告。

法律遵从性：系统的所有操作应符合所在地区的法律法规要求，包括但不限于电子商务、知识产权和消费者权益保护。

# 其他需求

## 开源许可证

软件采用GNU通用公共许可证（GPL）。

## 国际化需求

系统将以中文和英文两种语言提供界面和文档，以覆盖最广泛的用户群体。系统将采用支持 Unicode 字符集、双字节字符，以便未来添加更多语言支持。