**义眼盯真 —— 图像篡改检测系统**

软件设计文档

成员：刘信方，李康，王劲东，袁崇昕，张芳博

日期：2025/04/02

目录

[1. 引言 4](#_Toc194476067)

[1.1. 编写目的 4](#_Toc194476068)

[1.2. 项目范围 4](#_Toc194476069)

[1.3. 项目概述 4](#_Toc194476070)

[1.4. 参考文献 4](#_Toc194476071)

[1.5. 术语表 5](#_Toc194476072)

[2. 系统概述 6](#_Toc194476073)

[2.1. 产品视角与特性 6](#_Toc194476074)

[2.2. 用户类别和特征 6](#_Toc194476075)

[2.3. 操作环境 7](#_Toc194476076)

[2.4. 设计和实现约束 7](#_Toc194476077)

[2.5. 用户文档 7](#_Toc194476078)

[2.6. 假设与依赖 8](#_Toc194476079)

[3. 系统架构 9](#_Toc194476080)

[3.1. 架构设计 9](#_Toc194476081)

[3.2. 分解描述 10](#_Toc194476082)

[3.3. 设计依据 10](#_Toc194476083)

[4. 数据设计 11](#_Toc194476084)

[4.1. 数据描述 11](#_Toc194476085)

[4.2. 数据字典 13](#_Toc194476086)

[5. 组件设计 14](#_Toc194476087)

[5.1. 性能需求 14](#_Toc194476088)

[5.2. 安全与安保需求 15](#_Toc194476089)

[5.3. 质量需求 16](#_Toc194476090)

[6. 人机界面设计 17](#_Toc194476091)

[6.1. 用户界面概述 17](#_Toc194476092)

[6.2. 屏幕图像 17](#_Toc194476093)

[6.3. 屏幕对象和操作 17](#_Toc194476094)

[7. 需求矩阵 18](#_Toc194476095)

[8. 附录 19](#_Toc194476096)

# 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。

## 项目范围

**软件开发人员**：为程序员和工程师提供关键需求信息，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。

## 项目概述

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；

了解和描述软件实现所需的全部信息，为软件设计、确认和验证提供基准；

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。

## 参考文献

“义眼盯真 —— 图像篡改检测系统” 作为一款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。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。

可视化与报告功能：系统对图像中疑似篡改区域进行动态标注，方便用户快速定位和了解图像篡改情况。系统生成定制化检测报告，报告内容涵盖基本信息、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。同时，系统采用统一推理引擎（ONNX Runtime）实现跨平台兼容，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敏感信息进行不可逆加密，保障用户数据安全和隐私。

## 术语表

[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.

# 系统概述

## 产品视角与特性

在数字内容广泛传播的当下，AI 生成图像技术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，满足多个领域对图像真实性检测的迫切需求。

“义眼盯真——图像篡改检测系统” 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、图像检测、篡改可视化、报告生成、接口服务等。通过这些功能，本系统将为新闻媒体、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。

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。拥有最高级别的权限，负责系统的设置管理、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。

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。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。

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。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。

## 操作环境

在 Web 端，主流浏览器（Chrome、Firefox、Safari 等）均可访问系统。

服务器使用 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，同时使用 OpenCV 图像处理库实现图像的预处理和后处理。数据库采用 MySQL 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，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

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进行分类存储，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。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。

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

对软件进行全面概述，介绍主要功能、应用场景以及系统的优势（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。

技术文档

使用架构图和文字说明相结合的方式，详细描述系统的整体架构。对关键技术选型进行解释。通过数据库 ER 图和文字描述，展示系统所使用数据库的详细数据模型，帮助开发人员理解数据是如何存储和管理的。提供系统开放 API 接口的详细使用说明，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，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、图像处理库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。

硬件性能假设：假设开发过程中所使用的本地测试设备以及云计算资源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。

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不会发生重大变化，并且系统现有的数据脱敏技术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来补充训练数据，提高模型对特定类型图像篡改的识别能力。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。

# 系统架构

## 架构设计

用户界面接口是用户与 “义眼盯真 —— 图像篡改检测系统” 交互的关键部分，设计良好的用户界面接口对于提升用户体验至关重要。本系统将为用户提供美观、大方、直观且操作简单的具备现代风格的用户界面接口。

3.1.1 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。

3.1.2 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。

3.1.3 标准按钮、功能或导航链接

每个屏幕顶部设置统一的导航栏，包含 “首页”“图像检测”“历史记录”“我的账户” 等主要导航链接，方便用户在不同功能模块之间进行切换。

在表单和操作界面，设置 “上传”“检测”“保存报告”“取消” 等标准按钮，按钮命名清晰，操作明确。

3.1.4 错误信息显示标准

错误信息以清晰、具体、易懂的语言呈现，避免使用技术性术语，确保普通用户能够理解。

错误信息明确指出问题所在，并提供相应的解决建议或操作步骤，帮助用户快速解决问题。

错误信息以模态对话框的形式弹出显示，吸引用户的注意力，确保用户能够注意到并理解错误内容。

## 分解描述

3.2.1 阿里云 OSS 接口

规格说明：系统集成阿里云 OSS，用于存储用户上传的图像以及生成的检测报告。接口支持文件的上传、下载、删除和管理操作。上传文件时，接口会自动为文件分配唯一标识符，并记录文件的元数据。下载文件时，可根据文件的标识符快速定位并获取文件。同时，支持对文件进行分类管理，可根据用户、时间等维度对文件进行分组存储，方便后续的数据检索和管理。

版本号：使用阿里云 OSS 官方提供的最新稳定版本 SDK。

安全要求：接口采用阿里云 OSS 提供的安全机制，对文件的访问进行严格控制。所有数据传输均进行加密，防止数据在传输过程中被窃取或篡改。

3.2.2 OpenAI LLM 接口

规格说明：系统集成 OpenAI 大语言模型接口，用于辅助用户理解检测结果、提供相关知识和建议。当用户查看检测报告时，可通过该接口向 OpenAI 大语言模型发起询问，获取对检测结果的详细解读、图像篡改相关的技术原理、防范措施等信息。接口支持自然语言交互，用户能使用日常语言提问，系统会将问题传递给 OpenAI 大语言模型，并把其生成的回答返回给用户，还支持批量提问和多轮对话，以满足用户不同使用场景。

版本号：选用 OpenAI 市场上主流且性能稳定的大语言模型，使用其官方提供的最新 API 版本，以保证获得最新的模型能力和功能支持。

安全要求：在与 OpenAI 大语言模型交互过程中，严格遵守数据隐私保护原则，对用户的提问和 OpenAI 大语言模型的回答进行加密处理，确保数据安全性，同时对用户的提问内容进行过滤和审核，防止恶意提问或敏感信息泄露。。

## 设计依据

本系统无需额外硬件接口。

# 数据设计

## 数据描述

4.1.1 描述与优先级

该功能为用户提供登录和注册系统的服务，是用户使用系统其他功能的前提条件，优先级为高。它涉及普通用户、企业用户和系统管理员三类参与者，当用户尚未登录系统或需要注册新账户时，通过打开系统界面选择 “登录” 或 “注册” 功能来触发。

4.1.2 刺激 / 响应序列

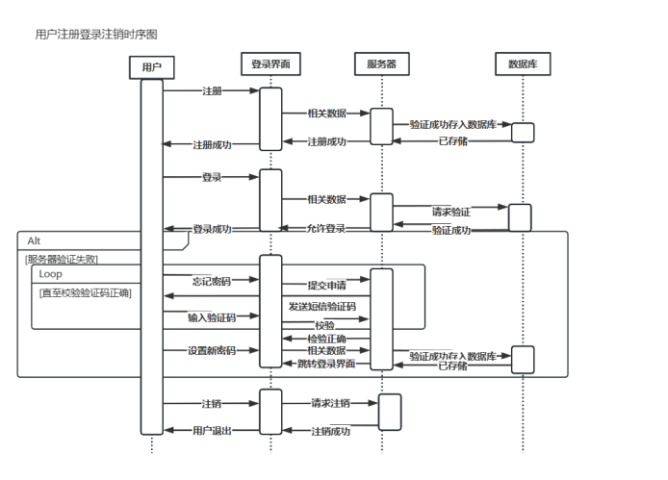
登录：用户选择登录方式（邮箱、手机号或第三方账号），输入对应账号和密码并点击 “登录”（刺激），系统调用相应 API 验证身份，验证成功则进入系统主界面（响应）；若账号或密码错误，系统提示 “账号或密码错误”（响应）；用户选择 “忘记密码”，系统引导用户通过绑定邮箱或手机号找回密码（响应）；若登录失败次数过多，系统锁定账户并提示用户稍后重试（响应）。

注册：用户在注册页面填写相关信息（刺激），系统验证信息合法性，验证成功则将相关数据存入数据库并提示注册成功（响应）。

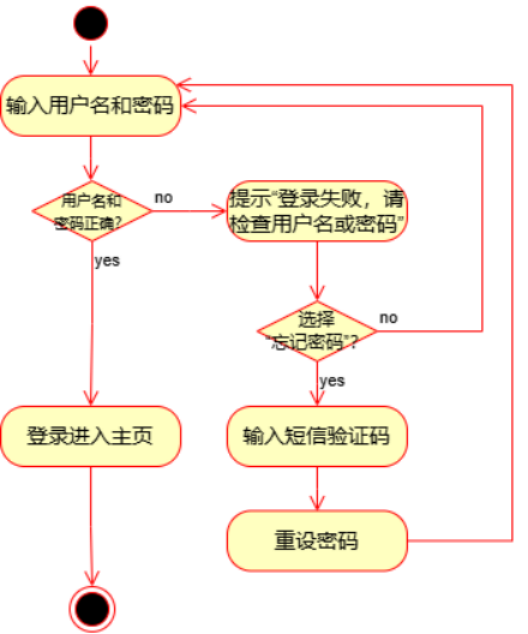
4.1.3 功能需求

REQ-1：支持邮箱、手机号和常见第三方账号（如微信、QQ）登录，集成对应 API 进行身份验证。

REQ-2：提供登录状态错误提示功能，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。

时序图



活动图

## 数据字典

4.2.1 描述与优先级

此功能允许已登录的普通用户和企业用户对图像进行真实性检测，对于保障图像内容的可靠性至关重要，优先级为高。当用户在系统中选择 “单图检测” 或 “批量检测” 功能时触发。

4.2.2 刺激 / 响应序列

用户进入检测页面，选择上传单张图像或批量上传图像（企业用户）（刺激），系统接收图像后进行预处理，调用基于 YOLO - 11n - seg 优化的深度学习模型进行检测，检测完成后生成检测报告并展示给用户（响应）；若上传图像格式不支持，系统提示 “文件格式不支持，请上传正确格式的图像”（响应）；检测过程中若出现异常（如模型加载失败），系统提示 “检测异常，请稍后重试”（响应）。

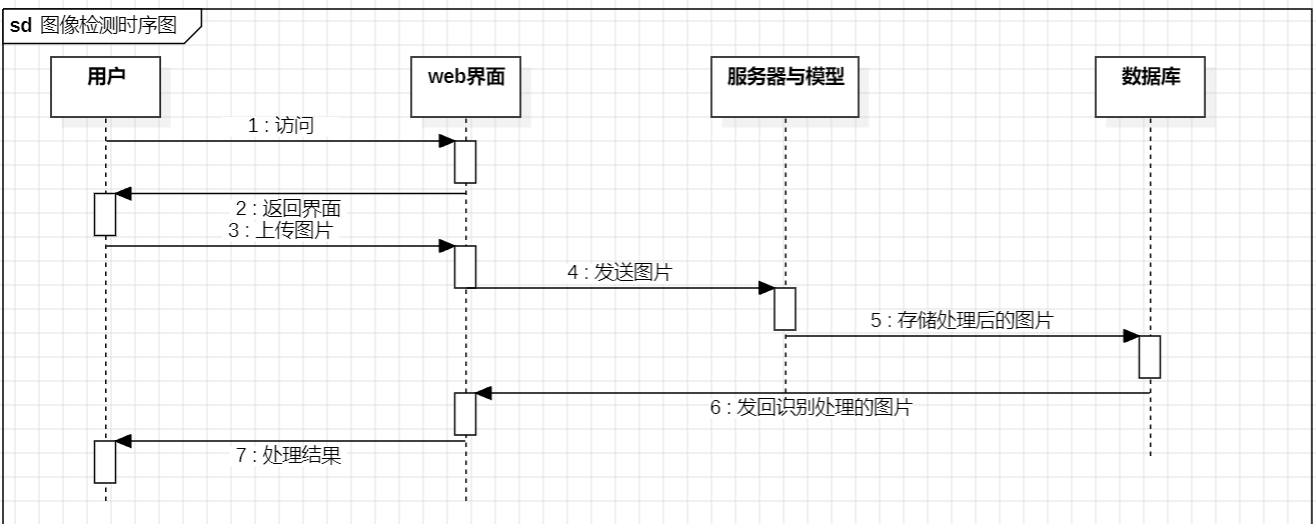
4.2.3 功能需求

REQ-4：支持单图和批量检测功能，批量检测支持一次上传 50 张图像。

REQ-5：系统自动识别图像格式，对不支持的格式进行提示。

REQ-6：利用优化的深度学习模型，实现对主流伪造手段（如 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等）的精准检测，检测准确率≥85%。

REQ-7：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、篡改区域定位、篡改置信度分析等，并支持多种格式（如 PDF、HTML）导出。



时序图

# 组件设计

## 性能需求

5.1.1 资源利用率

CPU利用率：移动端轻量化模型运行时，CPU利用率应≤30%；云端服务在高并发场景下单节点CPU利用率应≤70%，避免过载导致的响应延迟。

内存利用率：移动端模型内存占用≤50MB，边缘设备内存利用率≤80%；云端服务内存占用需动态优化，确保高并发下内存利用率≤75%。

存储利用率：检测数据定期自动清理，存储空间利用率保持在60%以下；模型训练数据存储采用分布式架构，冷热数据分层管理，存储利用率≤70%。

5.1.2 可扩展性

横向扩展能力：云端服务支持动态扩缩容，可自动增加GPU计算节点，峰值流量下扩展至≥5节点，确保吞吐量≥200张图像/分钟。

功能扩展能力：系统采用模块化架构，新增伪造检测类型时，可通过插件化设计集成，功能扩展耗时≤2人周。

5.1.3 容错性

故障检测与恢复：实时监控服务状态，检测到节点故障后10秒内切换至备用节点，服务中断时间≤30秒；模型推理异常时自动回滚至上一稳定版本，确保检测服务连续性。

数据备份与恢复：用户检测记录与模型参数每日备份至数据存储中心，保证数据丢失后恢复时间≤30分钟；数据集版本化管理，支持按需回滚至历史版本。

5.1.4 并发性

并发用户支持：支持≥500并发用户同时在线操作（包括图像上传、结果查看等）；视频检测任务队列支持≥50个并发处理线程。

请求处理能力：单节点处理能力：图像检测≥50请求/秒，视频检测≥10请求/秒（1080P分辨率）；高并发场景下响应时间波动范围≤±20%。

5.1.5 负载均衡

动态负载分配：采用加权轮询算法，根据服务器实时负载（CPU/内存）分配请求，节点间负载差异≤15%；视频检测任务按帧数拆分，均衡分配至多GPU节点。

流量监控与预警：实时监控API请求流量，流量峰值超过阈值时触发自动扩容，并推送告警至管理员。

5.1.6 安全性

抗攻击性能：在遭受DDoS攻击时，系统可维持≥50%的基础检测服务能力（如限制非关键功能）；恶意文件拦截率≥99%，误拦截率≤1%。

加密效率：AES-256加密对单张图像（1080P）的传输延迟影响≤0.3秒。

5.1.7 用户体验

响应时间：

|  |  |  |
| --- | --- | --- |
| **场景** | **设备/部署方式** | **响应时间上限** |
| 单张图像检测 | 移动端（轻量化模型） | ≤3秒 |
| 单张图像检测 | 云端（GPU加速） | ≤2秒 |
| 视频检测（1分钟时长） | 云端（批量处理） | ≤5分钟 |
| 批量检测（100张图像） | 云端集群 | ≤3分钟 |
| 批量检测（100张图像） | 云端集群 | ≤3分钟 |

页面加载时间：Web端主界面加载时间≤2秒，检测结果页加载时间≤1.5秒；移动端交互操作流畅度FPS≥60。

## 安全与安保需求

数据隐私保护：所有用户上传的数据在传输过程中需使用HTTPS协议加密，存储时采用AES-256加密算法，确保数据在传输和存储环节的安全性。用户数据仅用于检测分析，未经用户明确授权，禁止向第三方共享或用于其他用途。

访问控制：系统需实现基于角色的访问控制，区分普通用户、企业用户和管理员权限，限制敏感操作。用户身份验证需支持多因素认证，防止未授权访问。

抗攻击能力：系统需具备反恶意注入检测机制，防止上传恶意文件干扰服务。

API接口需设置速率限制（如单IP每秒最大请求数≤50），防范DDoS攻击。

## 质量需求

5.3.1 可靠性

系统稳定性：应保证99.9%的正常运行时间，排除计划维护和不可抗力因素。

故障恢复：发生故障时，系统应能在15分钟内自动恢复服务，或在30分钟内通过人工干预恢复。

5.3.2 可维护性

代码可读性：代码遵循编码规范，具有良好的注释和文档，便于理解和维护。

日志记录：日志系统记录完整操作轨迹（如用户请求、模型推理结果、异常事件），日志保留周期≥6个月，便于问题追踪和调试。

模块化设计：提供模块化设计，支持独立更新模型或界面，无需整体停机。

错误处理：系统的错误处理机制能够对异常情况进行捕获、记录和响应。

5.3.3 可扩展性

新增模块：支持新增伪造类型检测模块，通过插件化架构实现快速集成。

接口兼容：API接口兼容RESTful标准，支持第三方开发者自定义功能扩展。

# 人机界面设计

## 用户界面概述

软件采用GNU通用公共许可证（GPL）。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。

## 屏幕图像

系统将支持多种语言，以覆盖最广泛的用户群体，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，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。

## 屏幕对象和操作

遵守《中华人民共和国网络安全法》《个人信息保护法》等法律法规，确保数据采集、存储及处理流程合法合规。

系统需通过第三方安全审计，符合ISO/IEC 27001信息安全管理体系标准。

# 需求矩阵

# 附录