⚖ 智鉴
源代码相似性智能鉴定

快速开始
- 系统概述
- 功能特性
- 快速入门

操作指南
- 完整工作流程
  - 创建项目
  - 上传代码
  - 文件映射
  - 智能比对
  - 专家审查
  - 结论判定
  - 报告下载
  - 项目冻结

比对算法详解
- 算法体系概述

首页 / 用户手册

系统概述

智鉴是一款专业的源代码相似性智能鉴定系统，专为司法鉴定机构、知识产权保护、软件侵权诉讼等场景设计。系统结合了传统的文本比对算法与先进的人工智能技术，提供科学、客观、可追溯的代码相似性鉴定服务。

💡 **核心价值**
本系统不仅能够自动化完成代码比对工作，更重要的是提供**完整的鉴定依据链**，让每一个相似性判断都有据可查、有理可依，满足司法鉴定的严格要求。

适用场景

- **软件著作权侵权鉴定**：对比原告与被告软件源代码的相似程度
- **代码抄袭检测**：学术论文、开源项目的代码原创性检验
- **商业秘密泄露分析**：核心代码是否被非法复制使用
- **软件版权归属**：确定代码的原始作者和版权归属
- **开源协议合规**：检测是否违规使用开源代码

功能特性

多级比对算法
从普通行级到专家五级，6种比对模式覆盖不同精度需求，支持自定义分隔符配置。

AI智能分析
集成大语言模型，自动分析代码功能、识别对应关系、评估相似性原因。

可视化比对
并排对比视图，颜色标注差异，支持同步滚动，直观展示比对结果。

专业鉴定报告
自动生成符合司法鉴定要求的报告，包含完整比对数据和结论依据。

多租户隔离
支持多单位、多用户、多项目的数据完全隔离，确保数据安全。

灵活配置
比对参数、AI模型、排除规则均可自定义，适应不同鉴定需求。

快速入门

以下是使用本系统完成一次代码鉴定的基本步骤：

注册登录 → 创建项目 → 上传代码 → 智能比对 → 专家审查 → 结论判定 → 下载报告 → 项目冻结

⚠ **重要提示**
在正式开始鉴定前，请确保已正确配置大模型API（如需使用AI分析功能），并了解各比对级别的适用场景。

完整工作流程

本节详细介绍从项目创建到报告生成的完整操作流程。

创建项目

1. **进入项目列表**
   登录系统后，点击左侧导航栏的"项目管理"进入项目列表页面。
2. **点击"新建项目"**
   点击页面右上角的"新建项目"按钮，打开项目创建表单。
3. **填写项目信息**
   - **项目名称**：如"张三诉李四软件侵权案"
   - **案号**：法院立案案号或委托编号
   - **项目描述**：简要说明鉴定背景和目的

上传代码

项目创建成功后，系统会自动跳转到文件上传页面。

📁 **文件上传说明**

上传项	说明	支持格式
原告代码	主张权利方的源代码文件/文件夹	ZIP压缩包或直接拖拽文件夹
被告代码	被诉侵权方的源代码文件/文件夹	ZIP压缩包或直接拖拽文件夹

💡 **支持的代码格式**
系统支持主流编程语言：Python、Java、JavaScript/TypeScript、C/C++、C#、Go、Rust、PHP、Ruby、Swift等。配置文件（JSON、XML、YAML）和脚本文件（Shell、Batch）同样支持。

文件映射

文件映射是将原告代码文件与被告代码文件建立对应关系的过程。这是代码比对的基础。

映射类型

类型	说明	处理方式
同名相同	文件名相同，内容哈希值一致	直接判定为完全相同
同名相似	文件名相同，内容有差异	需要进行详细比对
不同名相似	文件名不同但内容相似	需要算法匹配后比对
独有文件	仅存在于一方的文件	可尝试算法匹配

算法匹配

对于不同名的文件，系统提供**算法匹配**功能，基于内容相似度自动发现潜在对应关系：

- 选择需要匹配的独有文件或不同名文件
- 设置相似度阈值（推荐 0.6-0.8）
- 选择比对模式（普通或专家级别）
- 执行算法匹配，系统自动建立映射关系

智能比对

智能比对是系统提供的全自动化比对流程，一键完成从AI功能分析到结论判定的全部工作。

智能比对流程

1. **AI功能分析**
   多线程分析所有代码文件的功能、创新性和技术难度。
2. **哈希与文件名匹配**
   基于文件哈希值和文件名自动建立初步对应关系。
3. **算法匹配分析**
   使用30%阈值进行内容相似度匹配，发现不同名但内容相似的文件。
4. **AI匹配核对**
   AI审核算法匹配结果，纠正错误匹配，寻找遗漏的对应关系。
5. **算法相似度计算**
   采用专家一级比对计算所有映射文件对的相似度。
6. **AI相似性分析**
   深度分析相似原因，给出相似性等级评估。
7. **更新比对结论**
   根据AI分析结果自动更新结论（高度相似+100%相似度=相同，否则=实质相同）。
8. **AI撰写鉴定结论**
   综合所有比对数据，AI自动撰写鉴定结论初稿。

☑ **智能比对优势**
- 全自动化执行，无需人工干预
- 支持从任意阶段重新开始
- 实时显示进度和处理状态
- 窗口支持最大化查看详细进度

专家审查

专家审查用于人工审核和调整智能比对的结果，确保鉴定结论的准确性。

审查操作步骤

1. 在项目列表点击"专家审查"进入详情页面
2. 查看比对统计数据，点击数字查看具体文件列表
3. 在代码比对页面筛选和查看文件对
4. 点击"详情"查看具体的代码差异对比
5. 在详情页面切换不同的比对级别（普通行级、专家1-5级）
6. 查看AI分析结果并调整结论

批量生成比对详情

选择多个文件后，可使用"批量生成比对详情"功能一次性生成指定级别的比对结果。

结论判定

系统支持对每个文件对设置鉴定结论：

AI相似性等级	结论映射规则	说明
高度相似	算法相似度=100% → **相同** 算法相似度<100% → **实质相同**	需结合算法相似度精确判定
相似	**基本相同**	代码主体结构一致，存在部分差异
部分相似	**部分相同**	存在相似片段，但也有独立部分
低度相似	**不同**	仅有少量相似特征
不相似	**不同**	无明显相似特征

⚠ **专家判断**
相似度数值仅作为参考依据，最终结论应由技术专家综合代码功能、业务逻辑、编程风格等因素进行判定。建议结合AI分析结果辅助决策。

报告下载

完成比对和结论判定后，可下载HTML格式的专业鉴定报告：

- 在项目列表点击"下载报告"按钮
- 系统自动生成包含完整比对数据的HTML报告
- 报告包含比对统计和代码比对两个页面
- 支持点击导航查看详细比对结果
- 报告可离线浏览，包含所有AI分析结果

💡 **报告内容**
- **比对统计**：文件映射类型统计、相似度分布
- **代码比对**：所有文件对的详细比对结果
- **AI分析**：功能分析、相似性分析结果
- **颜色标注**：与网页版一致的差异高亮显示

项目冻结

项目冻结用于保护已完成鉴定的项目数据不被修改。

冻结操作

- 在项目列表点击"冻结"按钮
- 系统会提示确认，因为冻结后项目将无法进行任何修改
- 确认后项目状态变为"已冻结"

解冻操作

- 只有**单位管理员**或**超级管理员**才有解冻权限
- 项目经理无法解冻自己创建的项目
- 解冻后项目恢复可编辑状态

⚠ **冻结注意事项**
冻结操作应在鉴定完成、报告定稿后进行。冻结后如需修改，必须联系有权限的管理员进行解冻。

比对算法体系概述

本系统采用**多级递进式**比对算法体系，从粗粒度到细粒度，满足不同场景的比对需求。算法设计遵循以下原则：

- **递进包含**：每一级别包含上一级的所有分隔符，逐级细化
- **语言无关**：基于文本分析，不依赖特定编程语言的语法解析
- **可追溯**：每个相似判断都有清晰的算法依据
- **可配置**：分隔符可在系统管理中自定义

📊 **算法分层架构**

```
┌─────────────────────────────────────────────┐
│              专家五级（最细粒度）              │
│   包含：运算符、括号、逗号、成员符、语句符、空白符   │
├─────────────────────────────────────────────┤
│                  专家四级                    │
│   新增：成员访问符（: :: . -> =>）              │
├─────────────────────────────────────────────┤
│                  专家三级                    │
│   新增：括号和逗号（( ) [ ] ,）                 │
├─────────────────────────────────────────────┤
│                  专家二级                    │
│   新增：语句分隔符（; { }）                     │
├─────────────────────────────────────────────┤
│                  专家一级                    │
│   新增：空格和制表符（\t 空格）                  │
├─────────────────────────────────────────────┤
│             普通行级（最粗粒度）               │
│   仅按行分割，使用 difflib 进行行级比对          │
└─────────────────────────────────────────────┘
```

比对级别详解

普通行级比对

属性	说明
级别标识	normal
分隔符	无（仅按换行符分割）
算法	Python difflib.SequenceMatcher
适用场景	快速初步筛查，判断代码是否存在大范围相似
优点	速度快，结果直观
缺点	无法检测行内的细微修改（如变量重命名）

示例：以下两行在普通模式下被判定为不同

```
// 原告代码
int result = calculateSum(a, b);

// 被告代码
int result = calculateSum( a,  b );
```

专家一级

属性	说明
级别标识	expert1
新增分隔符	`\t`（制表符）、`空格`
算法	Smith-Waterman 局部序列比对
适用场景	检测空格/缩进调整后的相似性
检测能力	可发现仅调整缩进或空格的代码复制

原理：将每行代码按空格和制表符拆分为词级Token，使用Smith-Waterman算法进行局部比对，能够识别：

- 缩进风格改变（Tab转空格、2空格转4空格等）
- 代码格式化工具造成的空白差异
- 手动添加/删除空格的微小调整

专家二级

属性	说明
级别标识	expert2
新增分隔符	`;`（分号）、`{`、`}`（花括号）
累计分隔符	空格、制表符、分号、花括号
适用场景	检测代码块边界、语句分隔的变化
检测能力	可发现语句合并/拆分、代码块重组

示例：以下代码在专家二级下会被识别为高度相似

```
// 原告代码
if (flag) { doSomething(); doOther(); }

// 被告代码
if (flag) {
    doSomething();
    doOther();
}
```

专家三级

属性	说明
级别标识	expert3
新增分隔符	`(`、`)`（圆括号）、`[`、`]`（方括号）、`,`（逗号）
适用场景	检测函数调用、参数列表的变化
检测能力	可发现参数顺序调整、参数增删、数组索引修改

示例：参数顺序变化检测

```
// 原告代码
result = calculate(x, y, z);

// 被告代码
result = calculate(y, x, z);  // 参数顺序调换
```

专家四级

属性	说明
级别标识	expert4
新增分隔符	`:`、`::`、`.`、`->`、`=>`
适用场景	检测对象成员访问、命名空间引用的变化
检测能力	可发现类/对象重命名、成员访问方式改变

示例：对象成员访问变化

```
// 原告代码 (C++)
std::vector::iterator it = vec.begin();

// 被告代码
Vector::iterator it = myVec.begin();  // 类名和变量名修改
```

专家五级

属性	说明
级别标识	expert5
新增分隔符	`+`、`-`、`*`、`/`、`=`、`!`、`<`、`>`、`&`、`|`、`%`、`^`
适用场景	最精细的比对，检测运算符级别的变化
检测能力	可发现运算符替换、表达式重写

💡 **级别选择建议**
- **初步筛查**：使用普通行级，快速过滤明显不相似的文件
- **一般鉴定**：使用专家二级或三级，平衡精度和效率
- **深度分析**：对关键文件使用专家四级或五级，发现精细修改

Smith-Waterman 算法

Smith-Waterman 算法是一种经典的**局部序列比对**算法，最初用于生物信息学中的DNA/蛋白质序列比对。本系统将其应用于代码Token序列的相似性分析。

算法原理

该算法使用动态规划方法，构建一个评分矩阵，找出两个序列之间的最优局部对齐。核心思想：

1. **Token化**：将代码按分隔符拆分为Token序列
2. **构建评分矩阵**：计算每对Token的匹配得分
3. **回溯最优路径**：找出得分最高的对齐方式
4. **计算相似度**：基于对齐结果计算相似度分数

评分参数

参数	说明	默认值
匹配得分	两个Token完全相同时的得分	+2
不匹配惩罚	两个Token不同时的惩罚	-1
空隙惩罚	插入或删除Token的惩罚	-1

算法优势

- **局部对齐**：即使整体不同，也能发现局部相似片段
- **容忍插入删除**：能够识别添加/删除代码后的相似性
- **位置无关**：代码片段位置移动不影响检测

相似度计算原理

系统使用多个维度计算代码相似度，最终综合得出整体相似度评分。

相似度维度

维度	计算方法	权重
文本相似度	基于Token序列对齐的相似度	主要指标
结构相似度	代码结构（函数、类、块）的相似程度	辅助指标
行级覆盖度	相同/相似行占总行数的比例	参考指标

计算公式

```
整体相似度 = (匹配Token数 × 2) / (总Token数A + 总Token数B)

其中：
- 匹配Token数：通过Smith-Waterman对齐后匹配的Token对数量
- 总Token数A/B：两个文件分别的Token总数
```

☑ **相似度解读**
- **90%+**：极高相似度，可能存在直接复制
- **70-90%**：高相似度，需要详细审查
- **50-70%**：中等相似度，可能存在参考
- **30-50%**：低相似度，需综合判断
- **30%以下**：较低相似度，可能为巧合

AI智能分析概述

本系统集成大语言模型（LLM），提供三种AI辅助分析功能，帮助技术专家更准确地进行鉴定判断。

AI匹配分析
分析不同名文件之间的对应关系，智能识别重命名、拆分、合并的代码文件。

AI相似性分析
深入分析代码相似原因，判断是抄袭、参考、巧合还是通用模式。

AI功能分析
解读代码功能和业务逻辑，帮助理解代码的实际用途。

AI匹配分析

当原被告代码中存在文件名不同但内容可能对应的情况时，AI匹配分析能够智能识别这种对应关系。

适用场景

- 文件被重命名（如 `utils.py` → `helper.py`）
- 一个文件被拆分为多个文件
- 多个文件被合并为一个文件
- 模块结构重组但功能相同

分析结果

AI会给出对应关系的判断依据，包括：

- 功能描述对比
- 核心逻辑相似点
- 命名模式分析
- 置信度评估

AI相似性分析

对已建立映射的文件对进行深度分析，解释相似性的原因和性质。

分析维度

维度	分析内容
相似原因	代码复制、参考借鉴、通用模式、第三方库
修改痕迹	变量重命名、格式调整、注释修改、逻辑微调
独创性分析	核心算法是否独创、实现方式是否独特
抄袭可能性	综合判断抄袭的可能性等级

AI功能分析

解读代码的功能和业务逻辑，并评估代码的创新性和技术难度。

输出内容

- **功能摘要**：用通俗语言描述代码功能
- **技术实现**：使用了哪些技术和算法
- **业务场景**：这段代码可能用于什么业务
- **复杂度评估**：代码的技术复杂程度

创新性分析

AI会评估代码的创新程度：

创新等级	特征描述
高度创新	独特的算法设计、创新的架构模式、新颖的问题解决方案
一般创新	在现有技术基础上有改进或优化
常规实现	采用通用的技术方案和标准实现方式

难度分析

AI会评估代码的技术难度：

难度等级	特征描述
高难度	复杂的算法实现、高级的系统架构、深度的领域知识要求
中等难度	需要一定的技术功底，涉及多个模块的协调
基础难度	常规的CRUD操作、简单的数据处理、基础的界面开发

鉴定理论基础

源代码相似性鉴定的理论基础来源于软件工程、知识产权法和司法鉴定科学的交叉领域。

软件著作权保护范围

根据《计算机软件保护条例》，软件著作权保护的是程序的**表达形式**，而非思想或算法本身。这意味着：

- **受保护**：具体的代码实现、程序结构、变量命名风格
- **不受保护**：通用算法思想、编程惯例、必要的表达方式

实质性相似判定

司法实践中判定代码侵权需要证明**实质性相似**，而非简单的相似度数值。判定因素包括：

1. **接触可能性**：被告是否有机会接触原告代码
2. **相似程度**：代码相似的范围和深度
3. **相似原因**：是否存在合理的独立创作解释
4. **抄袭痕迹**：是否存在刻意掩盖的痕迹

鉴定标准

本系统的鉴定标准参考了业界通行做法和司法鉴定规范。

相似度等级标准

等级	相似度范围	典型特征	鉴定建议
极高相似	≥ 90%	代码几乎完全相同，可能仅有格式差异	高度怀疑复制，需排除合理解释
高度相似	70% - 90%	主体结构相同，存在部分修改	需详细分析修改痕迹和原因
中度相似	50% - 70%	存在明显相似片段，也有不同部分	需结合功能分析综合判断
低度相似	30% - 50%	部分代码相似，整体差异较大	可能为参考或巧合
不相似	< 30%	无明显相似特征	一般不构成侵权

⚠ **重要说明**
相似度数值仅作为参考，不能作为唯一判定依据。实质性相似的判定需要综合考虑代码的创作性、相似部分的重要性、是否存在独立创作的合理解释等因素。

结论判定依据

判定流程

1. **量化分析**
   使用本系统计算相似度数值，获取客观数据支撑。
2. **细节审查**
   对高相似度文件进行人工审查，确认相似的具体内容。
3. **AI辅助分析**
   使用AI分析功能获取相似原因的专业解读。
4. **排除合理解释**
   分析是否存在独立创作、使用公共库、行业惯例等合理解释。
5. **综合判定**
   由技术专家综合所有证据，形成鉴定结论。

抄袭判定要素

以下特征可作为抄袭的判定依据：

- **相同的错误或缺陷**：如果两份代码存在相同的bug或不规范写法
- **无意义的相似**：如相同的无用代码、多余的空行模式
- **命名规律一致**：变量、函数命名遵循相同的个人风格
- **注释内容相似**：尤其是个性化的注释内容
- **结构完全一致**：代码组织方式、文件结构高度相同

用户与权限

本系统支持多租户架构，实现单位、用户、项目三级数据隔离。系统采用**基于角色的访问控制（RBAC）**机制，不同角色拥有不同的权限范围。

角色体系

系统定义了四种用户角色，按权限从高到低排列：

角色	权限层级	权限范围	典型用户
超级管理员	1（最高）	系统全部权限，包括单位管理、全局配置、所有项目	系统运维人员
单位管理员	2	本单位全部权限，用户管理、本单位所有项目	鉴定所负责人
项目经理	3	管理自己创建的项目，配置技术专家，授权项目	案件承办人
技术专家	4（最低）	参与被分配或被授权的项目的比对分析工作	技术鉴定人员

功能菜单权限

不同角色看到的系统菜单不同：

菜单项	超级管理员	单位管理员	项目经理	技术专家
鉴定项目	✓	✓	✓	✓
单位管理	✓	✗	✗	✗
用户管理	✓	✓	✗	✗
专家管理	✓	✓	✓	✗
专家审批	✓	✓	✓	✗
邮件配置	✓	✗	✗	✗
比对配置	✓	✓	✗	✗
LLM配置	✓	✓	✓	✗
个人中心	✓	✓	✓	✓

项目权限详解

项目权限分为**查看权限**和**管理权限**两类：

项目查看权限

允许用户查看项目信息、比对结果、下载报告：

角色	可查看的项目范围
超级管理员	所有单位的所有项目
单位管理员	本单位的所有项目
项目经理	自己创建的项目 + 被授权查看的项目
技术专家	被分配参与的项目 + 被授权查看的项目

项目管理权限

允许用户修改项目、执行比对、分配专家、授权项目：

角色	可管理的项目范围
超级管理员	所有单位的所有项目
单位管理员	本单位的所有项目
项目经理	仅自己创建的项目
技术专家	无管理权限（只能提交审查意见）

项目授权机制

项目授权功能允许项目所有者或管理员将项目访问权限授予其他用户。

谁可以授权

授权人角色	可授权的项目范围
超级管理员	所有单位的所有项目
单位管理员	本单位的所有项目
项目经理	仅自己创建的项目

可被授权的对象

只能授权给**同一单位**的以下角色：

- **项目经理**：同单位的其他项目经理
- **技术专家**：同单位的技术专家

权限类型

授权时可以选择两种权限类型：

权限类型	标识	说明
查看权限	查看	只能查看项目信息、比对结果、下载报告，不能进行修改操作
鉴定权限	鉴定	可以参与项目鉴定工作，如提交专家审查意见

💡 **授权操作入口**
在项目列表页面，点击项目操作栏的"授权"按钮，可以打开授权管理弹窗。在弹窗中可以：
- 查看已授权的用户列表（包含角色和权限类型）
- 为新用户授权（选择权限类型后点击"授权"按钮）
- 撤销已有用户的授权

项目冻结/解冻权限

操作	可执行的角色	说明
冻结项目	超级管理员、项目所有者	项目经理可以冻结自己的项目
解冻项目	超级管理员、单位管理员	项目经理**无法**解冻自己冻结的项目，需联系管理员

⚠ **冻结权限设计原因**
项目冻结后代表鉴定完成、数据定稿。为防止误操作或篡改，项目经理不能自行解冻，必须由更高权限的管理员审批解冻。这是数据安全的重要保障措施。

专家分配权限

将技术专家分配到项目参与鉴定工作的权限：

角色	可分配专家的项目范围
超级管理员	所有项目
单位管理员	本单位的所有项目
项目经理	仅自己创建的项目

可被分配的专家范围：已通过该单位审批的技术专家，以及该单位的项目经理（项目经理也可以作为专家参与项目）。

专家审批流程

技术专家注册后需要通过审批才能加入单位并参与项目鉴定。

审批流程

1. **专家注册**
   技术专家通过注册页面注册账号，选择要加入的单位。
2. **邮箱验证**
   专家收到验证邮件，点击链接完成邮箱验证。
3. **等待审批**
   专家申请进入待审批状态，等待单位管理员或项目经理审批。
4. **审批处理**
   审批人在"专家审批"页面查看申请，可以通过或拒绝。
5. **完成加入**
   审批通过后，专家正式加入单位，可被分配到项目参与鉴定。

审批权限

角色	可审批的范围
超级管理员	所有单位的专家申请
单位管理员	本单位的专家申请
项目经理	本单位的专家申请

专家管理

在"专家管理"页面，管理员和项目经理可以管理技术专家用户。

主要功能

- **添加专家**：手动创建专家账号（自动审批通过）
- **编辑专家**：修改专家的姓名、手机号、状态等信息
- **删除专家**：删除专家账号
- **发送验证邮件**：为邮箱未验证的专家重新发送验证邮件

💡 **发送验证邮件**
如果专家的邮箱显示"未验证"状态，可以点击操作栏的"发送验证"按钮重新发送验证邮件。专家收到邮件后点击链接即可完成验证。

比对配置

管理员可在"系统管理 → 比对配置"中自定义各级别的分隔符。

配置项说明

- **级别名称**：如"专家一级"、"专家二级"等
- **分隔符列表**：该级别使用的分隔符字符
- **启用状态**：是否在比对时可选
- **排序顺序**：在界面中的显示顺序

💡 **配置建议**
建议保持递进包含的原则，即高级别包含低级别的所有分隔符。这样可以确保分析结果的一致性和可比性。

大模型配置

AI分析功能需要配置大语言模型。系统支持多种LLM提供商，**首选推荐使用智鉴大模型**。

⭐ **首选推荐：智鉴大模型**
智鉴大模型是专为代码相似性鉴定场景优化的自研大模型，具有以下优势：
- **私有化部署**：可部署在客户内网环境，代码数据不外泄，完全满足司法鉴定的保密要求
- **专业优化**：针对代码比对、功能分析、相似性判断等场景深度优化，鉴定结果更准确
- **数据安全**：所有数据在本地处理，无需上传至第三方服务器，确保商业秘密和敏感代码的安全
- **离线运行**：支持完全离线环境部署，适用于涉密项目和内网隔离场景

如需部署智鉴大模型，请联系技术支持获取私有化部署方案。

模型管理

大模型配置页面位于"系统管理 → 大模型配置"，包含三个核心功能模块：

模型列表
管理系统中可用的大模型：

- **系统模型**：管理员配置的全局可用模型，所有用户可见
- **个人模型**：用户自己添加的模型，仅本人可见和使用
- 支持按供应商分组显示，方便管理
- 可配置模型的 API 地址、参数、超时时间等

API Key 管理

管理大模型调用所需的 API 密钥：

Key 类型	标识	说明
我的 Key	我的	用户自己添加的 API Key，仅本人可用
已授权 Key	已授权 或 系统授权	管理员授权给用户或单位使用的 Key
系统级 Key	系统级（待授权）	管理员创建但尚未授权的 Key（仅管理员可见）

💡 **API Key 使用优先级**
系统按以下优先级选择可用的 API Key：
1. 用户自己添加的 Key（优先使用）
2. 直接授权给用户的 Key
3. 授权给用户所在单位的 Key

注意：系统级 Key 必须通过授权才能被普通用户使用，新注册用户需要自行添加 Key 或联系管理员获取授权。

授权管理（管理员）

管理员可以将系统级 API Key 授权给指定的单位或用户：

- **授权给单位**：该单位下所有用户都可以使用此 Key
- **授权给用户**：仅指定用户可以使用此 Key
- 支持设置配额限制（Token 数量）
- 支持设置过期时间
- 可随时启用/禁用授权

API Key 优先级逻辑

当用户执行 AI 分析时，系统按以下优先级选择可用的 API Key：

📊 **API Key 选择流程**

```
┌─────────────────────────────────────────┐
│            用户发起 AI 分析请求            │
└─────────────────────────────────────────┘
                     ↓
┌─────────────────────────────────────────┐
│ Step 1: 检查用户自己添加的 Key              │
│ ├─ 查找用户个人配置的、启用状态的 API Key     │
│ └─ 如果找到 → 使用此 Key                   │
└─────────────────────────────────────────┘
                     ↓ 未找到
┌─────────────────────────────────────────┐
│ Step 2: 检查直接授权给用户的 Key            │
│ ├─ 查找管理员直接授权给该用户的 Key          │
│ ├─ 检查授权是否启用、未过期、配额未用完      │
│ └─ 如果找到 → 使用此 Key                   │
└─────────────────────────────────────────┘
                     ↓ 未找到
┌─────────────────────────────────────────┐
│ Step 3: 检查授权给用户所在单位的 Key         │
│ ├─ 查找授权给用户所属单位的 Key             │
│ ├─ 检查授权是否启用、未过期、配额未用完      │
│ └─ 如果找到 → 使用此 Key                   │
└─────────────────────────────────────────┘
                     ↓ 未找到
┌─────────────────────────────────────────┐
│ 结果：无可用 Key，提示用户配置或联系管理员获取授权 │
└─────────────────────────────────────────┘
```

各角色的 API Key 权限

角色	可添加自己的 Key	可使用被授权的 Key	可创建系统级 Key	可授权 Key 给他人
超级管理员	✓	✓	✓	✓
单位管理员	✓	✓	✓	✓（本单位范围）
项目经理	✓	✓	✗	✗
技术专家	✗（无 LLM 配置菜单）	✓（通过被分配的项目）	✗	✗

⚠ **技术专家的 AI 功能使用**
技术专家没有 LLM 配置菜单，无法自行添加 API Key。但当技术专家被分配到项目后，可以使用项目所有者（项目经理）配置的 Key 来执行 AI 分析功能。这确保了：
- 专家无需关心 Key 配置，专注于鉴定工作
- API 费用由项目经理或单位统一承担
- Key 的安全性得到保障（专家无法看到 Key 内容）

其他支持的模型提供商

除智鉴大模型外，系统也支持接入以下第三方大模型服务：

提供商	API地址	推荐模型	说明
阿里通义千问	https://dashscope.aliyuncs.com	qwen-max, qwen-plus, qwen-coder	国产大模型，代码理解能力强
DeepSeek	https://api.deepseek.com/v1	deepseek-coder, deepseek-chat	代码专用模型，性价比高
OpenAI	https://api.openai.com/v1	gpt-4, gpt-4-turbo	通用能力强，需境外访问
本地 Ollama	http://localhost:11434	codellama, deepseek-coder	本地部署，完全离线可用
其他兼容API	自定义	-	支持OpenAI兼容接口的模型

⚠ **数据安全提示**
使用第三方大模型服务时，代码内容会发送至对应服务商的服务器进行处理。如涉及商业秘密、敏感代码或有保密要求的鉴定项目，**强烈建议使用智鉴大模型私有化部署方案**或本地 Ollama 部署，确保数据安全。

常见问题

Q: 相似度很高就一定是抄袭吗？

A: 不一定。高相似度只是表象，需要分析原因。以下情况可能导致高相似度但不构成抄袭：

- 使用了相同的开源库或第三方组件
- 遵循了相同的行业标准或规范
- 实现了通用的算法或设计模式
- 代码由同一个人或团队开发

Q: 比对哪个级别最准确？

A: 没有"最准确"的级别，各级别适用于不同场景：

- 普通行级适合快速筛查
- 专家二至三级适合一般鉴定
- 专家四至五级适合精细分析

建议对重要文件使用多个级别分析，综合判断。

Q: AI分析结果可以作为鉴定依据吗？

A: AI分析结果是辅助参考，不能作为直接依据。最终鉴定结论应由人类专家综合各方面证据后作出。AI的价值在于提供分析思路和发现人工可能遗漏的细节。

Q: 如何处理二进制文件和资源文件？

A: 系统默认排除二进制文件（如.exe、.dll、.so）和资源文件（如图片、音视频）。这些文件不属于源代码范畴，不在比对范围内。可在比对配置中自定义排除规则。

术语表

术语	解释
Token	代码经过分隔符拆分后的最小单元，如变量名、运算符、关键字等
文件映射	将原告文件与被告文件建立对应关系的过程
实质性相似	法律概念，指两份作品在受著作权保护的表达上存在足够程度的相似
Smith-Waterman	局部序列比对算法，用于发现两个序列之间的最优局部对齐
哈希值	文件内容的数字指纹，相同内容产生相同哈希值
LLM	Large Language Model，大语言模型，如GPT、DeepSeek等
多租户	一套系统服务多个独立单位，数据完全隔离的架构模式

版本更新

v2.2.0 (2026年1月)

- ✅ **项目授权功能升级**：支持授权给项目经理和技术专家两种角色
- ✅ **双权限类型**：授权时可选择"查看权限"或"鉴定权限"
- ✅ **技术专家授权访问**：技术专家可以通过授权（非分配）方式访问项目
- ✅ **专家审批权限扩展**：项目经理现在可以审批本单位的技术专家申请
- ✅ **重新发送验证邮件**：专家管理页面支持为未验证邮箱的专家重新发送验证邮件
- ✅ **权限逻辑文档**：用户手册新增详细的权限逻辑说明
- ✅ **技术专家菜单精简**：技术专家登录后只显示"鉴定项目"和"个人中心"

v2.1.0 (2026年1月)

- ✅ **LLM 配置升级**：全新的模型中心化配置架构，支持灵活的模型和 Key 管理
- ✅ **用户自管理 API Key**：用户可添加和管理自己的 API Key，无需依赖管理员
- ✅ **授权管理**：管理员可将系统级 Key 精确授权给指定单位或用户
- ✅ **配额控制**：支持为授权设置 Token 配额限制和过期时间
- ✅ **用户自定义模型**：用户可添加自己的模型配置，满足个性化需求
- ✅ **权限隔离**：系统级 Key 默认不对普通用户可见，必须通过授权才能使用
- ✅ **ZIP 文件下载**：代码上传页面支持下载已上传的压缩包
- ✅ **MD5 值显示**：上传的压缩包显示 MD5 校验值，便于验证文件完整性

v2.0.0 (2026年1月)

- ✅ **智能比对**：全自动化8阶段比对流程，支持从任意阶段重新开始
- ✅ **专家审查**：重新设计的人工审查流程
- ✅ **AI创新性和难度分析**：评估代码的创新程度和技术难度
- ✅ **HTML报告下载**：可离线浏览的完整鉴定报告
- ✅ **项目冻结/解冻**：保护已完成项目数据的安全机制
- ✅ **批量生成比对详情**：选择多文件一次性生成比对结果
- ✅ **智能结论映射**：高度相似+100%相似度=相同，否则=实质相同
- ✅ **实时进度显示**：AI分析过程显示具体文件名和成功/失败状态
- ✅ **弹窗最大化**：智能比对进度窗口和统计详情窗口支持最大化
- ✅ **后端状态恢复**：服务重启后自动清理异常状态的项目

V1.0.0 (2025年12月)

- ✅ 基础代码比对功能
- ✅ 多级专家比对算法
- ✅ AI智能分析集成
- ✅ 多租户数据隔离
- ✅ 专业鉴定报告生成
- ✅ 用户权限管理
- ✅ 邮箱验证注册