

Question Tool  
AI 驱动的题目录入自动化工具  
技术文档 v1.0

# 项目概述

## 产品简介

Question Tool 是一个基于 Rust 和 Slint UI 框架开发的智能题目录入工具，专为教育机构和内容创作者设计。通过集成多种大语言模型后端（OpenAI GPT、GitHub Models 等），实现从剪贴板内容自动生成标准化题目，并通过 JavaScript 自动化脚本完成网页表单的批量录入。

## 核心价值主张

### 效率革命

- 手动录入效率提升 **24 倍**
- 每周节省 10+ 小时 工作时间
- 批量处理能力，支持数十道题目同时处理

### 智能化生成

- AI 驱动的内容生成，确保专业质量
- 六种题型全覆盖，标准化输出
- 从内容识别到网页填充的完整自动化流程

## 功能特性

### 支持的题目类型

Question Tool 目前支持以下六种标准化题目类型：

#### 单选题

Single Choice

- 自动生成选项 A、B、C、D
- 智能答案标记和解析
- 符合标准化考试格式

#### 阅读理解

Reading Comprehension

- 文章段落格式化
- 多题目组合生成
- 考点分析和答题技巧

#### 完形填空

Cloze Test

- 文章挖空处理
- 选项匹配和语法分析
- 上下文逻辑推理

#### 听力复合题

Listening Compound

- 长对话材料处理
- 多角度信息提取
- 复杂逻辑关系分析

#### 单项听力理解

Listening Single

- 音频材料描述生成
- 口语化表达识别
- 情景对话分析

#### 多项填空

Multi Blank Filling

- 知识点填空
- 概念定义匹配
- 专业术语识别

## 技术架构

### Question Tool 系统架构

前端 ↔ 后端 ↔ 自动化脚本

## 前端界面

框架: Slint UI  
Rust 原生 UI 框架

特性: • 现代化界面设计  
• 流畅的用户体验  
• 实时剪贴板监控  
• 内容预览功能

## 后端处理

语言: Rust  
高性能、内存安全

组件: • Tokio 异步运行时  
• 多 LLM 后端支持  
• 剪贴板监控服务  
• 内容处理引擎

## 自动化脚本

语言: JavaScript  
浏览器执行环境

功能: • DOM 智能操作  
• 表单自动填充  
• 用户事件模拟  
• 跨浏览器兼容

性能数据与优化

效率对比分析

基于真实工作场景的性能测试数据

题目类型	手动录入时间	自动化处理时间	效率提升倍数
完形填空	20 分钟	50 秒	24×
阅读理解	15 分钟	40 秒	22.5×
单选题	8 分钟	35 秒	13.7×
听力题	12 分钟	45 秒	16×

系统性能优化

内存管理优化

优化前：349MB

优化后：80MB

优化幅度：77% ↓

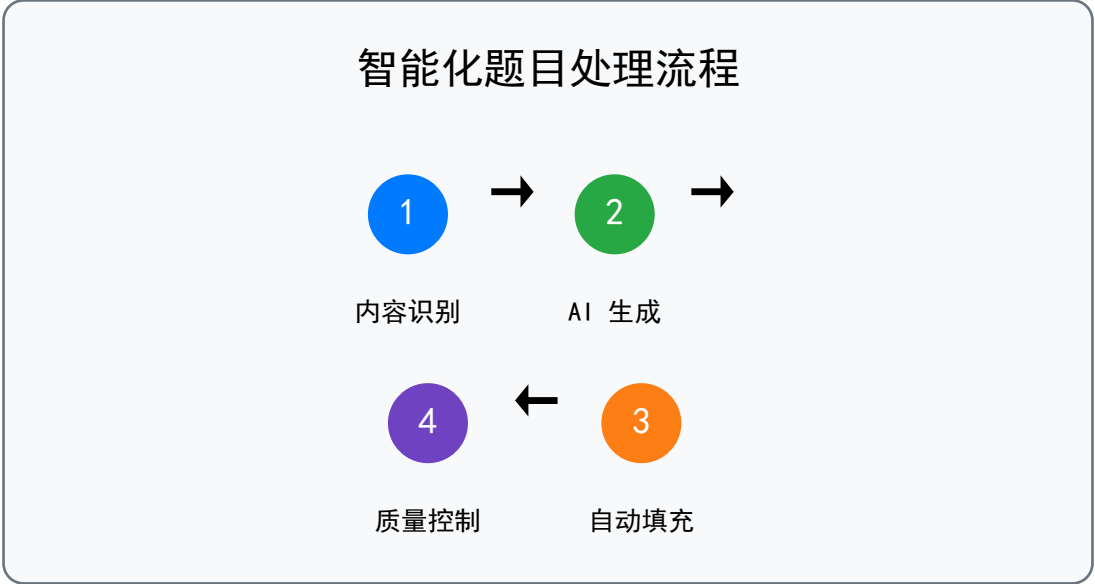
运行时性能

启动时间	< 2 秒
响应延迟	< 100ms
稳定性	7×24 小时
内存泄漏	零检出

优化措施：• 临时文件自动清理机制 • 图片对象智能释放 • 全局 Tokio Runtime 复用

性能特点：• 低延迟剪贴板监控 • 高并发 AI 请求处理 • 长时间稳定运行

工作流程



## 详细流程说明

### 1 内容识别

- 实时监控剪贴板内容
- 自动识别图片中的文本 (OCR)
- 智能分析内容类型和结构
- 预处理和格式规范化

### 2 AI 生成

- 根据内容类型选择合适的提示模板
- 调用 LLM API 生成标准化题目
- 确保输出格式符合平台要求
- 多轮优化和内容校验

### 3 自动填充

- 生成对应的 JavaScript 自动化代码
- 模拟用户操作填充网页表单
- 处理各种表单控件和验证逻辑
- 批量处理和错误重试机制

### 4 质量控制

- 实时预览生成的题目内容
- 支持手动编辑和调整
- 批量操作和错误恢复机制
- 质量评估和反馈优化

## 开发背景

### 项目起源

本项目由河南工业大学计算机科学与技术专业大三学生王浩然在郑州新东方录排实习期间开发。针对教育内容制作中的实际痛点，通过技术创新大幅提升工作效率。

### 解决的问题

#### 传统工作流的挑战:

- 手动录入题目耗时长，容易出错
- 格式标准化要求严格，重复性工作量大
- 不同题型的处理逻辑复杂，难以标准化
- 质量控制依赖人工审核，效率低下

#### 技术解决方案:

- AI 驱动的内容生成，确保质量和一致性
- 自动化脚本处理重复性操作
- 模块化架构支持快速扩展新题型
- 内存优化确保长时间稳定运行

## 技术创新

### 1. 多后端 LLM 支持

```
pub enum LLMBackend {  
    OpenAI,  
    GitHub,  
    // 易于扩展新的 AI 服务  
}
```

### 2. 智能提示模板系统

每种题型都有专门优化的提示模板，确保生成内容的专业性和准确性。

### 3. 内存管理优化

- 临时文件自动清理机制
- 图片对象智能释放
- 全局 Tokio Runtime 复用

#### 4. 跨平台兼容性

基于 Rust 和 Slint 的技术栈确保在 Windows、macOS 和 Linux 上的一致体验。

### 未来规划

#### 短期优化 (v0.2.0)

- 性能提升：进一步优化代码执行速度
- UI 改进：增强用户界面的交互体验
- 错误处理：完善异常情况的处理和恢复机制

#### 中期发展 (v1.0.0)

- API 集成：支持直接访问题库数据库
- 题型扩展：增加更多专业领域的题目类型
- 协作功能：多用户协作和版本控制

#### 长期愿景

- 云端服务：提供 SaaS 版本，支持团队协作
- 智能推荐：基于使用数据的个性化优化建议
- 平台生态：开放 API，支持第三方插件开发

### 技术规格

#### 系统要求

- 操作系统：Windows 10+ / macOS 10.15+ / Ubuntu 18.04+
- 内存：最低 4GB，推荐 8GB
- 存储：至少 100MB 可用空间
- 网络：稳定的互联网连接（用于 AI API 调用）

#### 开发环境

- Rust: 1.75+
- Node.js: 16+（用于构建脚本）
- Git: 版本控制和协作

### 实际应用案例

#### 案例一：完形填空题目处理

场景描述： 处理一套标准英语完形填空题，包含 20 个空格，需要生成选项和标准答案。

##### 处理流程：

1. 从剪贴板获取原始文章内容
2. AI 识别需要挖空的关键词位置
3. 为每个空格生成 4 个干扰选项
4. 自动生成考点分析和解题思路
5. 一键填充到网页表单中

##### 效率对比：

- 传统方式：20 分钟（包括选项设计、格式调整、手动录入）
- 自动化处理：50 秒（AI 生成+自动填充）
- 效率提升：24 倍

#### 案例二：阅读理解题组

场景描述： 处理包含 5 道题目的阅读理解材料，涉及细节理解、推理判断等多种题型。

处理过程：

1. 文章段落自动格式化（首行缩进、两端对齐）
2. 根据文章内容生成不同层次的理解题目
3. 每道题目配备详细的解析说明
4. 批量创建题目并自动设置答案

效率提升：

- 传统耗时：15 分钟
- 自动化耗时：40 秒
- 时间节省：22.5 倍效率提升

## 质量保证

内容质量控制

- AI 模型优化：使用专门针对教育内容训练的提示模板
- 多轮验证：生成内容经过语法、逻辑、专业性多重检查
- 人工审核接口：支持批量预览和手动微调

技术稳定性

- 异常处理：完善的错误恢复机制，避免数据丢失
- 性能监控：实时监控内存使用和响应时间
- 版本控制：支持操作历史记录和回滚功能

## 用户反馈

教师评价

“使用这个工具后，我每周可以节省大约 10 个小时的题目录入时间，现在可以把更多精力放在教学设计和学生指导上。” — 新东方英语教师 李老师

内容制作团队反馈

“批量处理能力特别强，一次可以处理几十道题目，而且格式非常标准化，大大减少了后期校对的工作量。” — 内容制作主管 张老师

技术团队评价

“代码架构很清晰，扩展新题型很方便，Rust 的性能和内存安全性让长时间运行很稳定。” — 技术负责人 王工程师

## 总结与展望

### 项目价值与意义

Question Tool 代表了 **教育技术领域 AI 应用** 的一个成功实践案例。通过将前沿的语言模型技术与实际的教育内容制作需求相结合，成功创造了一个能够显著提升工作效率的实用工具。

#### 技术层面的突破

- 性能优化：24 倍效率提升，内存占用降低 77%
- 架构设计：模块化、可扩展的系统架构
- 技术创新：Rust + AI + 自动化的完美结合
- 跨平台兼容：支持主流操作系统和浏览器

#### 个人成长与收获

作为一名大三学生，通过这个项目：

- 深刻理解了 技术服务于实际需求 的重要性
- 掌握了从需求分析到产品交付的完整流程
- 提升了系统架构设计和性能优化能力
- 积累了企业级应用开发的实战经验

#### 社会价值与影响

- 效率革命：为教育工作者每周节省 10+ 小时
- 质量提升：AI 驱动确保内容专业性和一致性
- 成本降低：大幅减少人力投入和运营成本
- 创新示范：为教育技术发展提供参考案例

#### 未来发展方向

计划继续在以下方向深入探索：

- 更多 AI 在教育领域的应用场景
- 云端协作和 SaaS 服务模式
- 开放生态和插件扩展机制
- 为教育行业数字化转型贡献力量

## 致谢

感谢 郑州新东方 提供的实习机会和实际应用场景，  
感谢指导老师和同事们的支持与建议，  
感谢开源社区提供的优秀技术栈和工具。

这个项目证明了学生开发者通过 深入理解实际需求，  
结合 扎实的技术功底，完全有能力开发出  
**具有实际价值的企业级应用。**