

# AI Tutor 前端 GUI 应用需求文档

---

## 1. 项目概述

### 1.1 项目背景

AI Tutor 是一个智能教学辅助系统，目前提供了一套功能完善的命令行界面（CLI）工具，用于管理班级、学生、作业和成绩。为了提升用户体验和操作效率，我们计划开发一个图形用户界面（GUI）应用，将 CLI 工具的所有功能集成到直观易用的可视化界面中。

### 1.2 项目目标

开发一个现代、简洁、信息密度高的 GUI 应用，实现 AI Tutor CLI 工具的所有核心功能，为教育工作者提供更加友好和高效的操作体验。

### 1.3 目标用户

- 教师
- 教学管理人员
- 教育技术专员

## 2. 功能需求

### 2.1 系统配置

- ☐ API 网关地址配置
- ☐ 系统连接测试功能

### 2.2 班级管理

- ☐ 创建班级（支持从 CSV/JSON 文件批量导入）
- ☐ 查看班级列表
- ☐ 查看班级详情（包括班级内的学生列表）

### 2.3 学生管理

- ☐ 添加学生到班级（支持从 CSV/JSON 文件批量导入）
- ☐ 查看班级学生列表

### 2.4 作业管理

- ☐ 创建作业
- ☐ 查看班级作业列表
- ☐ 批量批改作业（触发批改流程）
- ☐ 生成并打印反馈报告
- ☐ 生成成绩统计分析

### 2.5 数据导入/导出

- ☐ 支持从 CSV/JSON 文件导入班级和学生数据
- ☐ 支持将班级、学生、作业数据导出为 CSV/JSON 格式

2.6 模板生成

- ☐ 生成班级数据模板 (CSV/JSON)
- ☐ 生成学生数据模板 (CSV/JSON)

3. 用户界面设计

3.1 设计原则

- **简洁易用**：界面布局清晰，操作流程直观
- **信息密度高**：在有限的空间内展示更多有效信息
- **现代审美**：采用现代化的设计语言，视觉效果美观
- **响应式设计**：适配不同屏幕尺寸的设备

3.2 界面原型

3.2.1 主界面

- 顶部导航栏：系统标题、API 状态指示、配置入口
- 左侧边栏：功能模块导航（班级管理、学生管理、作业管理、数据管理）
- 主内容区：根据导航选择显示对应功能模块

3.2.2 班级管理界面

- 班级列表展示（ID、名称、创建时间）
- "创建班级"按钮，支持文件上传
- 点击班级条目可查看班级详情和学生列表

3.2.3 学生管理界面

- 选择班级后展示该班级的学生列表（ID、姓名、学号、创建时间）
- "添加学生"按钮，支持文件上传

3.2.4 作业管理界面

- 选择班级后展示该班级的作业列表（ID、标题、知识点、创建时间）
- "创建作业"表单
- 选择作业后可进行"批量批改"、"生成反馈报告"、"生成统计分析"操作

3.2.5 数据管理界面

- 提供数据导入、导出、模板生成功能的统一入口
- 明确标识各功能的操作对象（班级、学生、作业）

4. 技术要求

4.1 前端框架

- 推荐使用现代前端框架，如 React、Vue.js 或 Angular
- 优先考虑成熟稳定、社区活跃的技术栈

## 4.2 UI 组件库

- 使用现代化的 UI 组件库，如 Ant Design、Element UI、Material-UI 等
- 确保组件的美观性和交互体验

## 4.3 与后端通信

- 通过 HTTP RESTful API 与 AI Tutor 后端服务通信
- 支持自定义 API 地址配置

## 4.4 文件处理

- 支持 CSV 和 JSON 文件的解析与生成
- 提供文件上传和下载功能

# 5. 非功能性需求

## 5.1 性能

- 界面响应迅速，操作流畅
- 大数据量展示时需考虑分页或虚拟滚动

## 5.2 兼容性

- 主流现代浏览器（Chrome、Firefox、Safari、Edge）
- 支持 Windows、macOS、Linux 操作系统

## 5.3 安全性

- 数据传输安全（建议使用 HTTPS）
- 文件上传安全校验

## 5.4 可维护性

- 代码结构清晰，注释完整
- 遵循前端开发最佳实践

# 6. 项目交付

## 6.1 交付内容

- 完整的前端 GUI 应用源代码
- 详细的项目文档（包括开发文档、部署文档、用户手册）
- 可执行的应用程序包或部署说明

## 6.2 验收标准

- 实现所有功能需求

- 通过主要浏览器兼容性测试
- 代码符合质量标准
- 提供完整的文档