



编程教育与竞赛管理平台 - 需求规格说明书

1. 项目概述

- 核心目标：**构建一个集课程学习、作业提交、编程竞赛、智能测评于一体的综合性在线教育平台。
- 技术架构：**
 - 开发模式：**前后端分离 (Decoupled Architecture)
 - 前端技术：**Vue.js (构建单页面应用) + HTML5 + CSS3 + Echarts (可视化)
 - 后端技术：**Python (Django REST Framework)
 - 数据库：**MySQL
 - 架构类型：**B/S (Browser/Server) 架构
 - AI/算法支持：**百度/阿里人脸识别API、AI大模型接口（如ChatGPT/文心一言）

2. 核心创新点 (Highlights)

- 生物特征识别：**集成**人脸识别技术**，支持用户（学生/教师/管理员）通过刷脸完成安全登录。
- AI 智能助教：**内置**AI助手**，为学生提供7x24小时的编程问题咨询、代码除错建议及学习路径规划。
- 数据可视化驾驶舱：**利用 **Echarts** 图表库，为管理员和教师提供直观的数据看板（如作业提交率趋势、成绩分布饼图、竞赛参与度折线图等）。

3. 用户角色与功能详解

本系统包含三种用户角色：**管理员、教师、学生**。

3.1 管理员 (Administrator)

拥有系统最高权限，负责基础数据维护与全局监控。

- 基础管理**
 - 登录：**账号密码登录 / 人脸识别登录。
 - 个人中心：**修改密码、个人信息维护。
 - 看板数据统计：****[核心]** 使用Echarts展示全站数据（用户活跃度、新增课程数、竞赛参与人数等）。
 - 系统管理：**轮播图管理、系统日志、权限分配。
- 用户与班级管理**
 - 学生/教师管理：**增删改查用户信息、重置密码。
 - 班级管理：**创建班级、分配学生与教师。
- 教务与资源管理**
 - 课程类型/竞赛类型管理：**维护分类字典。
 - 课程信息管理：**审核或管理所有课程内容。

- **编程竞赛管理**：统筹安排竞赛场次、时间。
 - **竞赛报名管理**：查看全站报名情况。
 - **考评与成绩管理**
 - **作业/提交管理**：监控作业发布与提交情况。
 - **测评试卷/试题管理**：维护公共题库、试卷库。
 - **测评记录/成绩管理**：查看全站成绩，导出报表。
 - **错题本管理**：查看学生高频错题（用于教学分析）。
-

3.2 教师 (Teacher)

负责教学内容的生产、活动的组织及学生的考核。

- **教学管理**
 - **添加课程信息**：上传视频、课件、编写课程大纲。
 - **发布公告**：向班级发布通知（考试提醒、放假通知）。
 - **试卷/试题管理**：录入编程题、选择题，组建考试试卷。
 - **作业与考核**
 - **发布作业**：布置课后练习，设置截止时间。
 - **批改作业**：在线查看学生提交的代码或文档，进行评分和评语。
 - **学生成绩管理**：录入/修改学生成绩，查看班级成绩分布（Echarts可视化）。
 - **竞赛运营**
 - **举办编程竞赛**：发起新的竞赛，配置规则。
 - **报名审核**：审核学生参加特定竞赛的资格。
 - **互动与反馈**
 - **在线回复**：回答学生的课程提问。
 - **消息通知**：接收系统或学生的提醒。
-

3.3 学生 (Student)

系统的最终使用者，进行学习、练习与测评。

- **访问与个人中心**
 - **注册/登录**：支持人脸录入与识别登录。
 - **个人中心**：资料修改、头像上传。
- **学习与互动**
 - **课程信息浏览**：搜索课程，支持点赞、收藏、评论。
 - **在线沟通**：与教师私信或在评论区留言。
 - **AI助手咨询**：**[核心]** 唤起AI对话框，询问编程语法、逻辑问题。
 - **消息/公告浏览**：查看教师通知及回复。
- **考核与演练**
 - **编程竞赛浏览**：查看正在进行或往期的竞赛（支持搜索、排序）。

- **竞赛报名**：提交参赛申请。
- **竞赛考试**：在线答题（倒计时、代码提交）。
- **提交作业**：上传作业文档或直接提交代码。
- **错题本浏览**：查看自己做错的题目，进行针对性复习。
- **数据记录**
 - **成绩浏览**：查看课程成绩及竞赛排名。
 - **考试记录浏览**：回溯过往的测评详情。

4. 数据库实体初步规划 (ERD建议)

为了支撑上述功能，建议设计以下核心数据表：

- 用户表 (Users)**: 存储ID、账号、密码、角色、**人脸特征数据**。
- 课程表 (Courses)**: 课程名、教师ID、类型、视频地址、点赞数、收藏数。
- 竞赛表 (Competitions)**: 竞赛名、开始时间、结束时间、规则。
- 题目表 (Questions)**: 题目内容、类型（编程/选择）、答案/测试用例。
- 作业/试卷表 (Assignments/Papers)**: 关联课程或竞赛。
- 提交记录表 (Submissions)**: 学生ID、题目ID、提交内容、**AI评分/教师评分**、提交时间。
- 互动表 (Interactions)**: 评论、收藏、点赞记录。

5. 开发建议与路线图

- 第一阶段（基础搭建）**：完成Django+Vue环境配置，实现用户表设计及基本的登录注册（先做普通登录）。
- 第二阶段（核心业务）**：完成课程管理、作业发布与提交、基础的增删改查。
- 第三阶段（竞赛与题库）**：实现试题录入、在线考试逻辑（这是难点，涉及倒计时和自动判分逻辑）。
- 第四阶段（创新集成）**：接入人脸识别SDK，对接AI大模型API，集成Echarts图表。