**✅ 项目搭建总结与开发准备须知**

**📌 项目简介**

**项目名称：** classroom-interaction-system  
一套用于实现“学生提问 + 教师管理 + AI智能分类”的 Web 全栈系统，包含前后端、数据库、AI 模块。

**🧱 项目结构概览**

classroom-interaction-system/

├── frontend/ # 前端项目

│ ├── student-app/ # 学生端 (Vue2)

│ └── teacher-app/ # 教师端 (Vue2)

├── backend/ # 后端项目 (Express.js)

├── ai-service/ # AI 分类服务（关键词分类规则）

├── database/ # 数据库结构 & 测试数据

├── docs/ # 文档

└── README.md

**🧹 技术栏 & 依赖版本**

**✅ 前端（Vue）**

| **技术** | **版本** | **说明** |
| --- | --- | --- |
| Vue CLI | @vue/cli v5.0.8 | 项目创建脚手架 |
| Vue.js | 2.x | 使用 Vue 2，适配现有生态 |
| vue-router | 3.x | 配合 Vue 2 使用的路由管理 |
| axios | ^1.6.x | 前端调用后端 API |
| Vuex | ^3.x | 状态管理 |

✅ 推荐插件（VS Code）：

* **Vetur**：Vue 语法高亮 & 支持
* **Vue VSCode Snippets**：模板快速生成
* **Auto Rename Tag** / **Bracket Pair Colorizer 2**：标签闭合提示

**✅ 后端（Node.js + Express）**

| **技术/模块** | **版本** | **说明** |
| --- | --- | --- |
| Node.js | >=16.x | 推荐 Node 16 或以上 |
| Express.js | ^4.18.2 | Web 服务框架 |
| MySQL2 | ^3.x | 数据库连接 |
| cors | ^2.x | 解决跨域问题 |
| helmet | ^7.x | HTTP 安全头防护 |
| bcryptjs | ^2.x | 密码加密 |
| jsonwebtoken | ^9.x | JWT 登录认证 |
| nodemon | ^3.x（dev） | 开发时自动重启后端服务 |

**✅ 数据库（MySQL）**

* 使用 Navicat 或 MySQL Workbench 管理
* 数据库名：classroom\_system
* 提供 schema.sql 用于快速建表

**✅ AI 分类服务**

* 使用 Node.js 本地模块进行关键词分类
* 位于 ai-service/ 目录
* 可升级为 Python 服务或 ML 模型

**📁 文件夹说明（开发前务必熟悉）**

| **路径** | **说明** |
| --- | --- |
| frontend/student-app/src | 学生端页面、组件、路由 |
| frontend/teacher-app/src | 教师端页面、组件、路由 |
| backend/routes/ | 后端接口定义 |
| backend/models/ | 数据模型封装 |
| backend/services/ | 业务逻辑，含 AI 分类服务 |
| backend/middleware/ | JWT 验证、请求验证中间件 |
| backend/config/database.js | MySQL 连接配置 |
| ai-service/classifier.js | 关键词匹配规则 |
| database/schema.sql | MySQL 表结构定义 |
| docs/ | 包含 API 文档与部署说明 |

**⚠️ 开发注意事项**

1. **Vue 2 和 Vue 3 不兼容，务必统一版本**
   * 所有 Vue 项目都使用 Vue 2
   * vue-router@3 适配 Vue 2，vue-router@4 是 Vue 3 用的
2. **后端使用 CommonJS 模块语法**
   * 使用 require() 和 module.exports，不要混用 ESModules
3. **统一端口**
   * 学生端：8080
   * 教师端：8081
   * 后端服务：3000
4. **数据库连接信息**
   * 默认读取 backend/config/database.js
5. **注意路径大小写与模块导入一致**
   * Windows 对路径不敏感，Linux 服务器是敏感的

**🔄 常用开发命令**

# 启动学生端

cd frontend/student-app

npm run serve

# 启动教师端

cd frontend/teacher-app

npm run serve

# 启动后端服务

cd backend

npm run dev

# 启动 AI 服务（如果是独立服务）

cd ai-service

node classifier.js

**✅ 下一步建议**

可按优先级依次开发：

1. 🔐 用户认证系统（注册、登录）
2. 📝 学生端提交问题（API + 页面）
3. 📋 教师端查看问题、调用 AI 分类
4. 🤖 分类逻辑与关键词匹配
5. 🔔 实时推送 WebSocket（高级功能）

**其他有待补充….**