

茯苓在线教育平台建设

详细设计说明书

V1.0

产品名称：	茯苓在线教育平台
开发团队：	白虹贯日
队 长：	蔡博凯
队 员：	谢欣欧
队 员：	李兆煜
指导教师：	钱忠

文档修订记录

编号	版本号	修订时间	修订内容	修订人	审核人
1	V1.0	2024.7.15	初始创建	蔡博凯	蔡博凯

第 1 章 文档介绍	1
1.1 文档目的	1
1.2 文档范围	1
1.3 读者对象	1
第 2 章 总体设计	2
2.1 平台服务	2
2.2 用例图	3
第 3 章 数据设计	4
3.1 用户数据	4
3.1.1 类图设计	4
3.1.2 接口设计	4
3.1.3 数据库设计	4
3.2 教育机器人数据	5
3.2.1 类图设计	5
3.2.2 接口设计	5
3.2.3 数据库设计	5
3.3 心理测试数据	6
3.3.1 类图设计	6
3.3.2 接口设计	6
3.3.3 数据库设计	6
3.4 实时英文对话数据	7
3.4.1 类图设计	7
3.4.2 接口设计	7
3.4.3 数据库设计	7
3.5 教学 ppt 生成	7
3.5.1 类图设计	7

3.5.2 接口设计	7
3.5.3 数据库设计	7
3.6 教学视频生成	8
3.6.1 类图设计	8
3.6.2 接口设计	8
3.6.3 数据库设计	8
第 4 章 界面详细设计	9
4.1 用户注册	9
4.1.1 界面设计	9
4.1.2 交互逻辑	10
4.2 用户登录	11
4.2.1 界面设计	11
4.2.2 交互逻辑	12
4.3 教育机器人	13
4.3.1 界面设计	13
4.3.2 交互逻辑	14
4.4 心理测试	15
4.4.1 界面设计	15
4.5 实时英文对话	17
4.5.1 界面设计	17
4.5.2 交互逻辑	18
4.6 教学 ppt 生成	19
4.6.1 界面设计	19
4.6.2 交互逻辑	20
4.7 教学视频合成	21
4.7.1 界面设计	21
4.7.2 交互逻辑	22

4.8 用户管理	23
4.8.1 界面设计	23
4.8.2 交互逻辑	24
第 5 章 核心功能设计	25
5.1 信息校验	25
5.2 JWT	25
5.3 信息加密	27
5.4 websocket 连接	27
5.5 视频合成	28
第 6 章 模块详细设计	29
6.1 用户模块	29
6.2 教育机器人模块	31
6.3 心理测试模块	32
6.4 实时英文对话模块	34
6.5 教学 ppt 生成模块	35
6.6 教学视频合成模块	38
第 7 章 技术设计	40
7.1 技术栈	40

第 1 章 文档介绍

1.1 文档目的

本详细设计文档旨在详细描述茯苓在线教育平台的详细设计思路 and 具体实现的功能，为读者提供明确的技术指导和参考。

1.2 文档范围

本文档将涵盖平台服务、类图设计、界面设计、模块设计以及技术设计，确保所有参与者对平台的开发过程有详细的理解。

1.3 读者对象

本文档的读者对象是茯苓在线教育平台的开发者以及其他需要了解产品详细设计思路的人员。

第 2 章 总体设计

2.1 平台服务

平台提供的服务如表 2.1 所示。

表 2.1 服务介绍

服务名称	名称	运行端口	说明
数据库服务	mysql	3306	存储数据
用户服务	userservice	8081	基础服务
nacos 服务	nacos	8848	服务注册
gateway 服务	gatewayservice	10010	网关
教育机器人服务	helperservice	8091	业务服务
心理测试服务	docservice	8101	业务服务
实时英文对话服务	virtualservice	8131	业务服务
教学 ppt 生成服务	pptservice	8111	业务服务
教学视频合成服务	videoservice	8121	业务服务
前端服务 1	front	8080	运行前端页面
前端服务 2	front	5173	运行前端页面

2.2 用例图

图 2.2 介绍了平台涉及到的相关用户，图 2.3 介绍了平台的具体功能。



图 2.2 相关用户

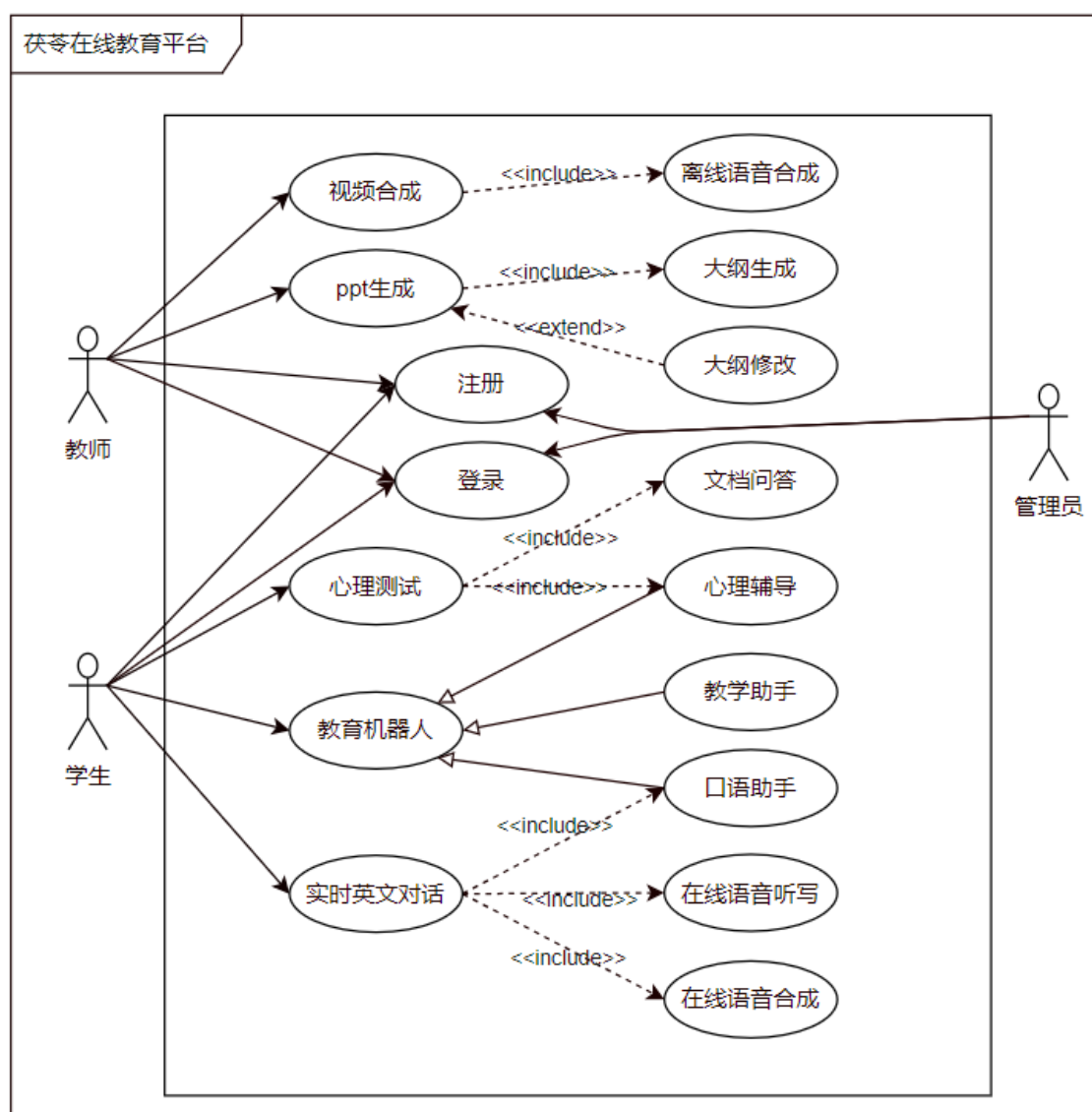


图 2.3 用例图

第 3 章 数据设计

3.1 用户数据

3.1.1 类图设计

有关用户数据的类图说明如图 3.1 所示。

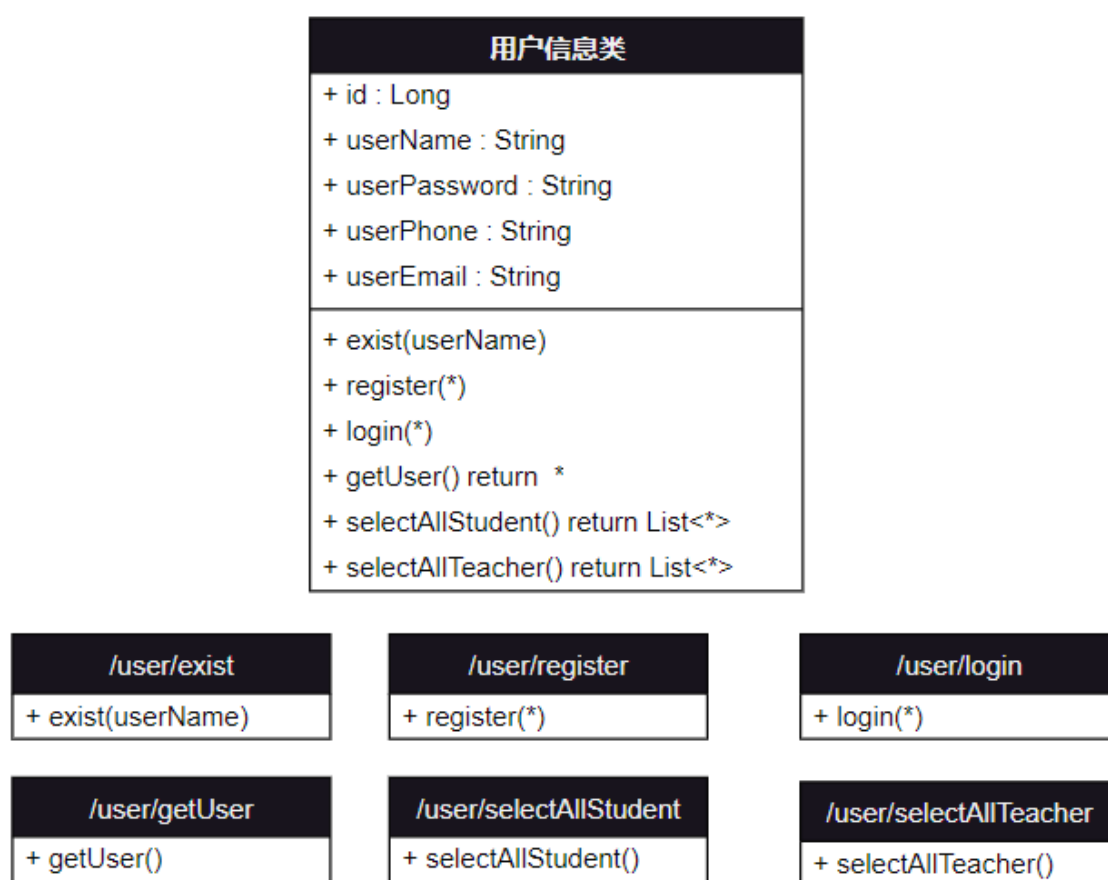


图 3.1 用户数据类

3.1.2 接口设计

见概要设计说明书 5.1.2。

3.1.3 数据库设计

见需求规格说明书 6.1。

3.2 教育机器人数据

3.2.1 类图设计

有关教育机器人的类图说明如图 3.2,所示。

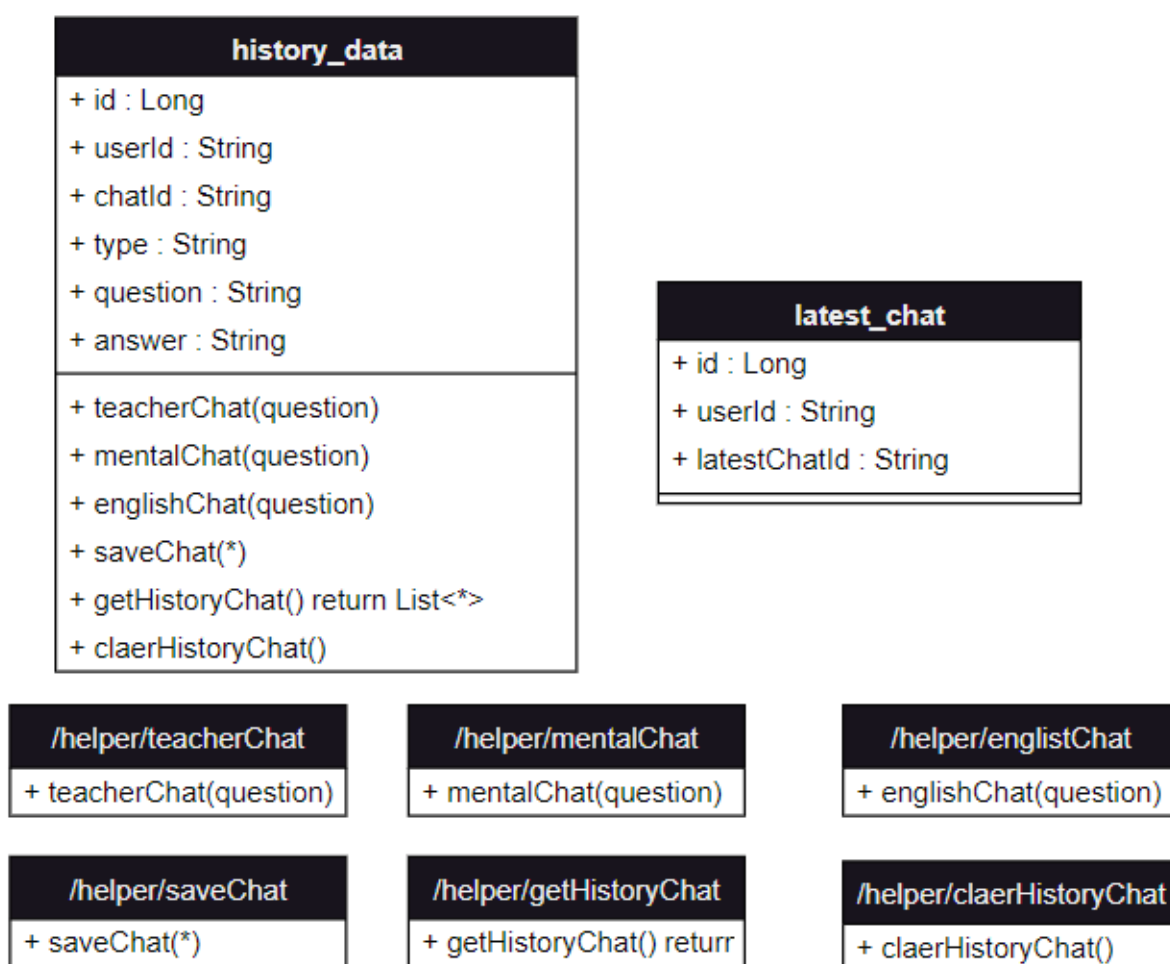


图 3.2 教育机器人类

3.2.2 接口设计

见概要设计说明书 5.3.2。

3.2.3 数据库设计

见需求规格说明书 6.2。

3.3 心理测试数据

3.3.1 类图设计

有关心理测试的类图说明如图 3.3 所示。

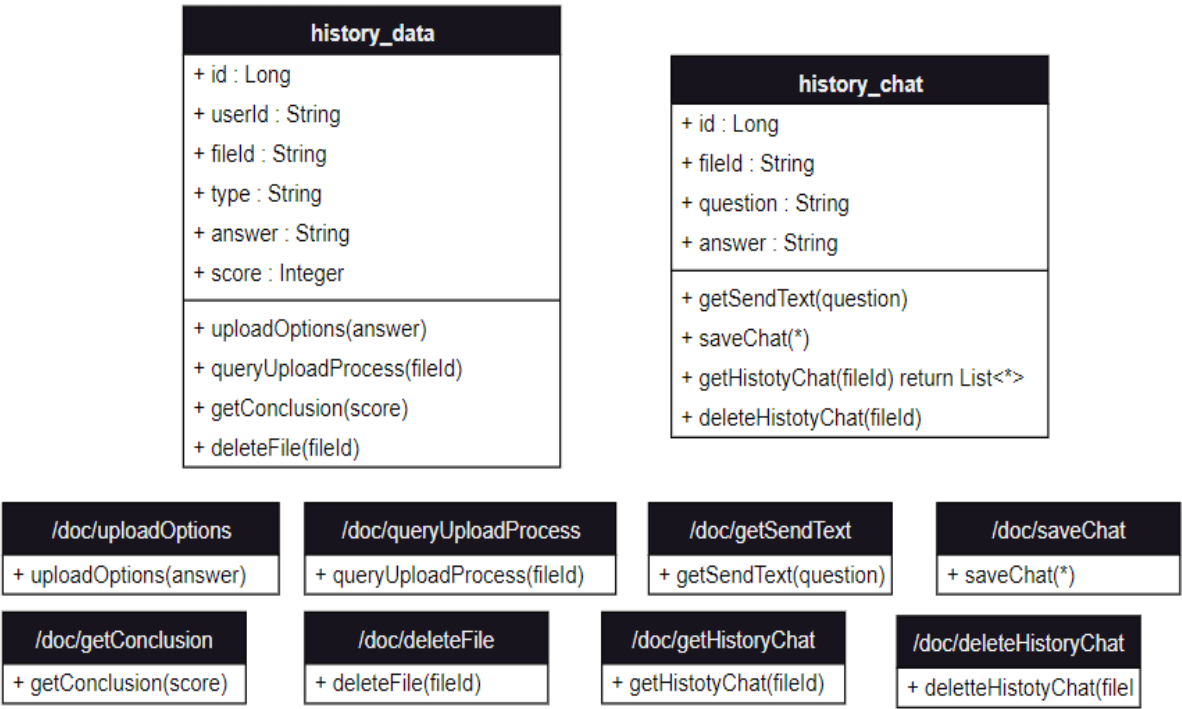


图 3.3 心理测试类

3.3.2 接口设计

见概要设计说明书 5.4.2。

3.3.3 数据库设计

见需求规格说明书 6.3。

3.4 实时英文对话数据

3.4.1 类图设计

由于实时英文调用了教育机器人模块的口语助手功能，所以数据类与教育机器人数据类似，这里不再赘述。

3.4.2 接口设计

见概要设计说明书 5.5.2。

3.4.3 数据库设计

见需求规格说明书 6.4。

3.5 教学 ppt 生成

3.5.1 类图设计

有关 ppt 生成的类图说明如下图 3.4 所示。

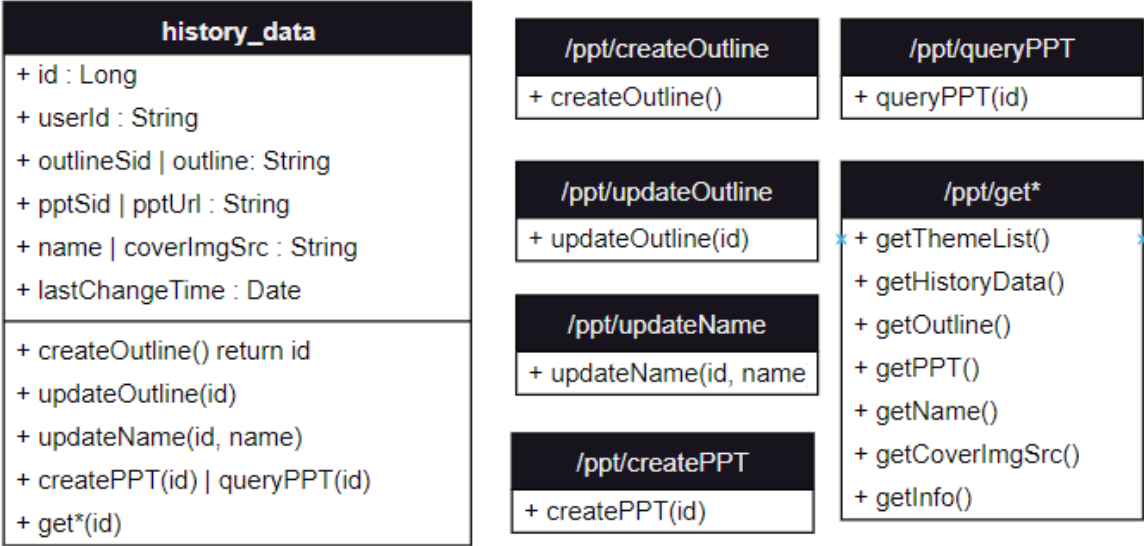


图 3.4 教学 ppt 生成类

3.5.2 接口设计

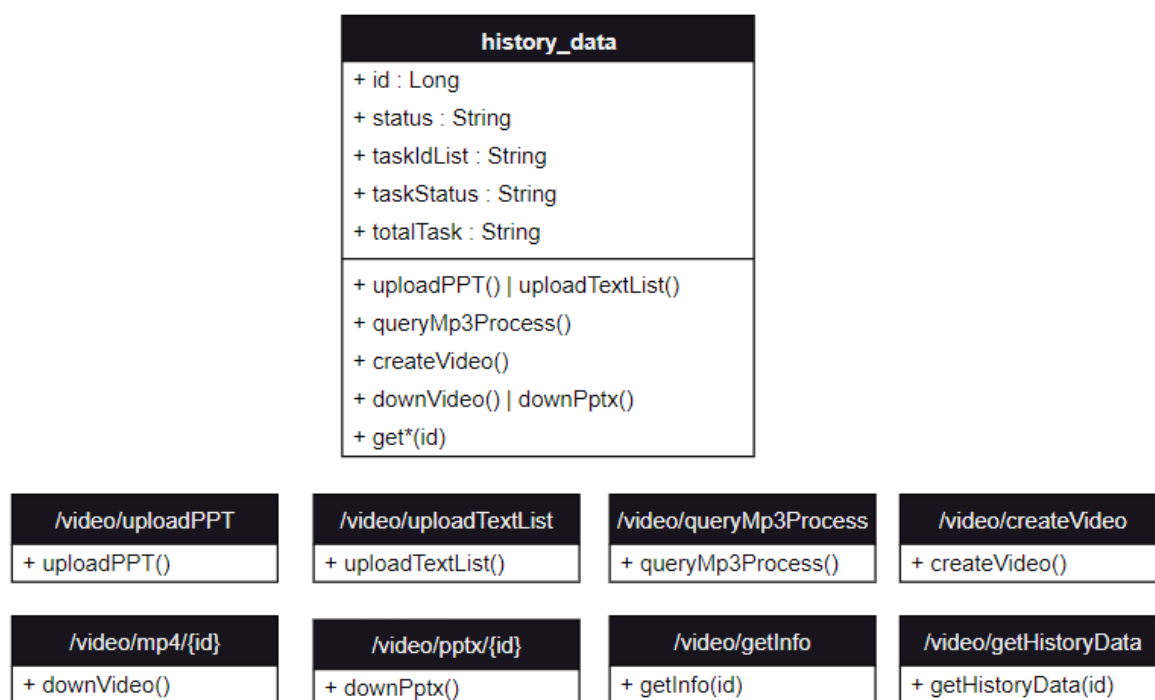
见概要设计说明书 5.6.2。

3.5.3 数据库设计

见需求规格说明书 6.5。

3.6 教学视频生成

3.6.1 类图设计



3.6.2 接口设计

见概要设计说明书 5.7.2。

3.6.3 数据库设计

见需求规格说明书 6.6。

第 4 章 界面详细设计

4.1 用户注册

4.1.1 界面设计

注册界面设计如图 4.1。

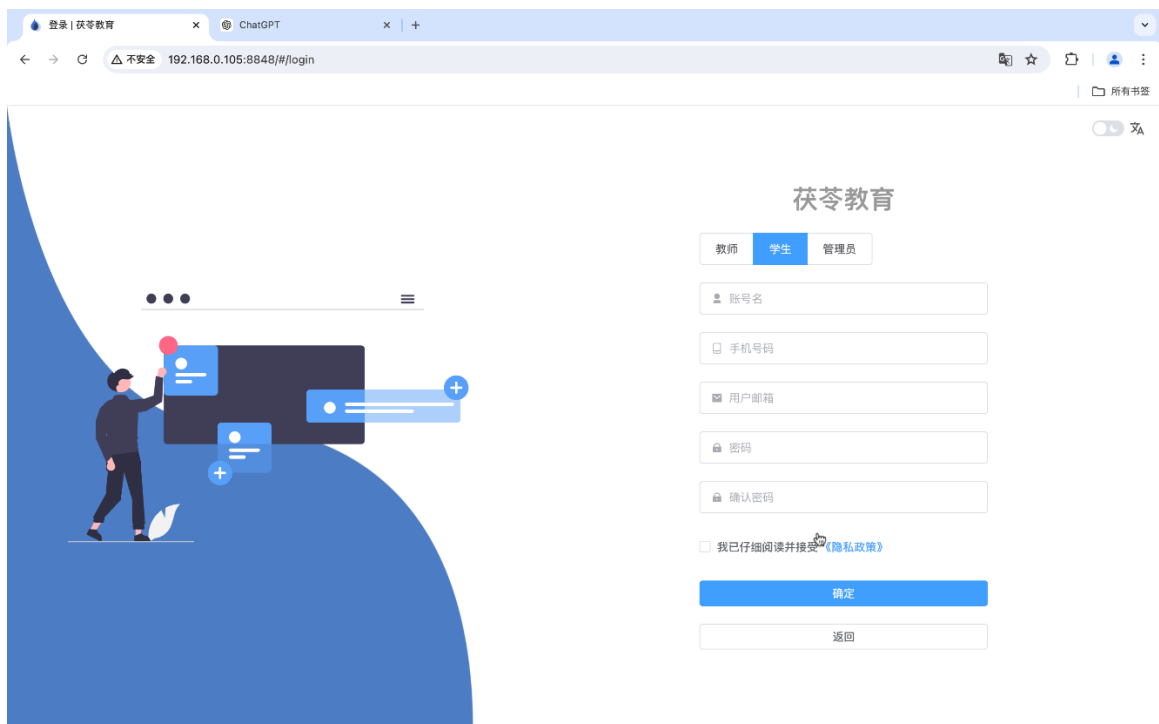


图 4.1 注册界面

4.1.2 交互逻辑

注册交互逻辑如图 4.2。

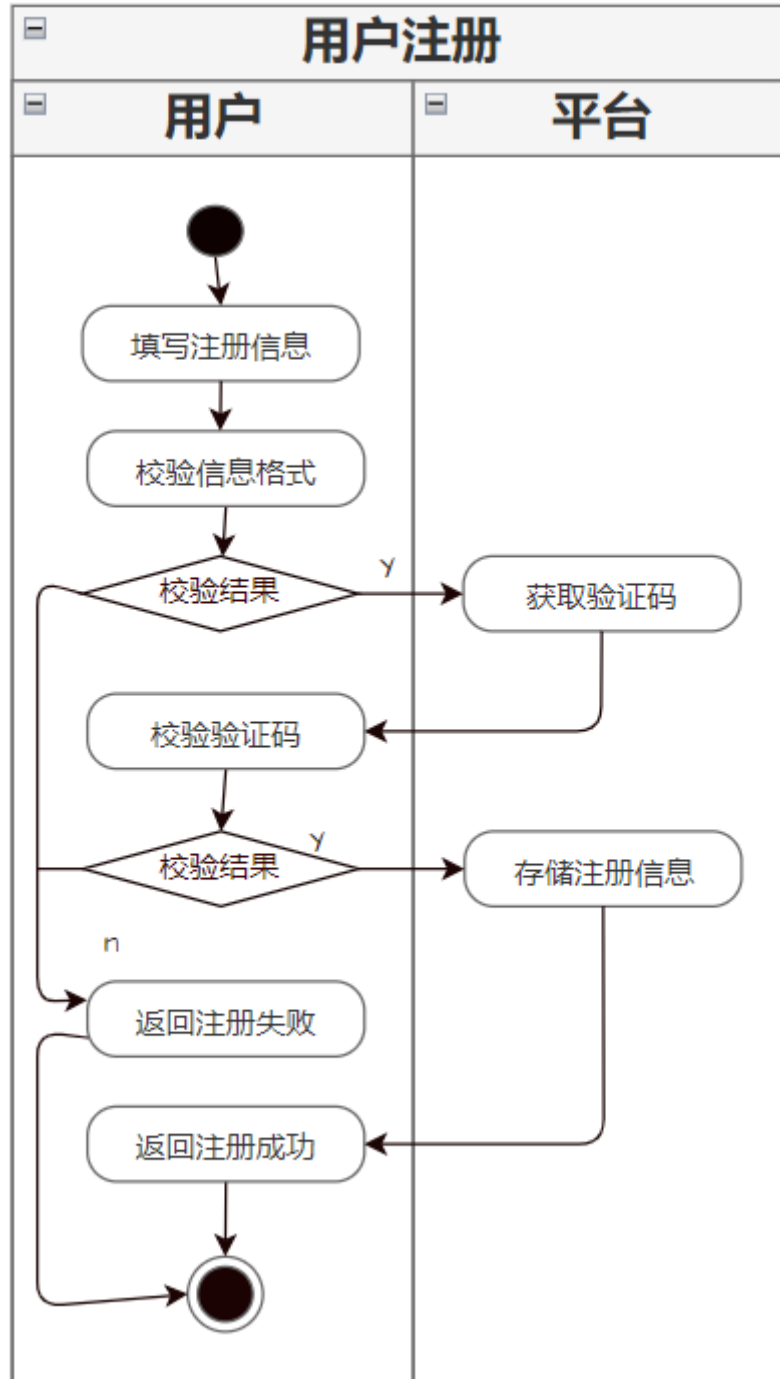


图 4.2 注册交互逻辑

4.2 用户登录

4.2.1 界面设计

登录界面设计如图 4.3。

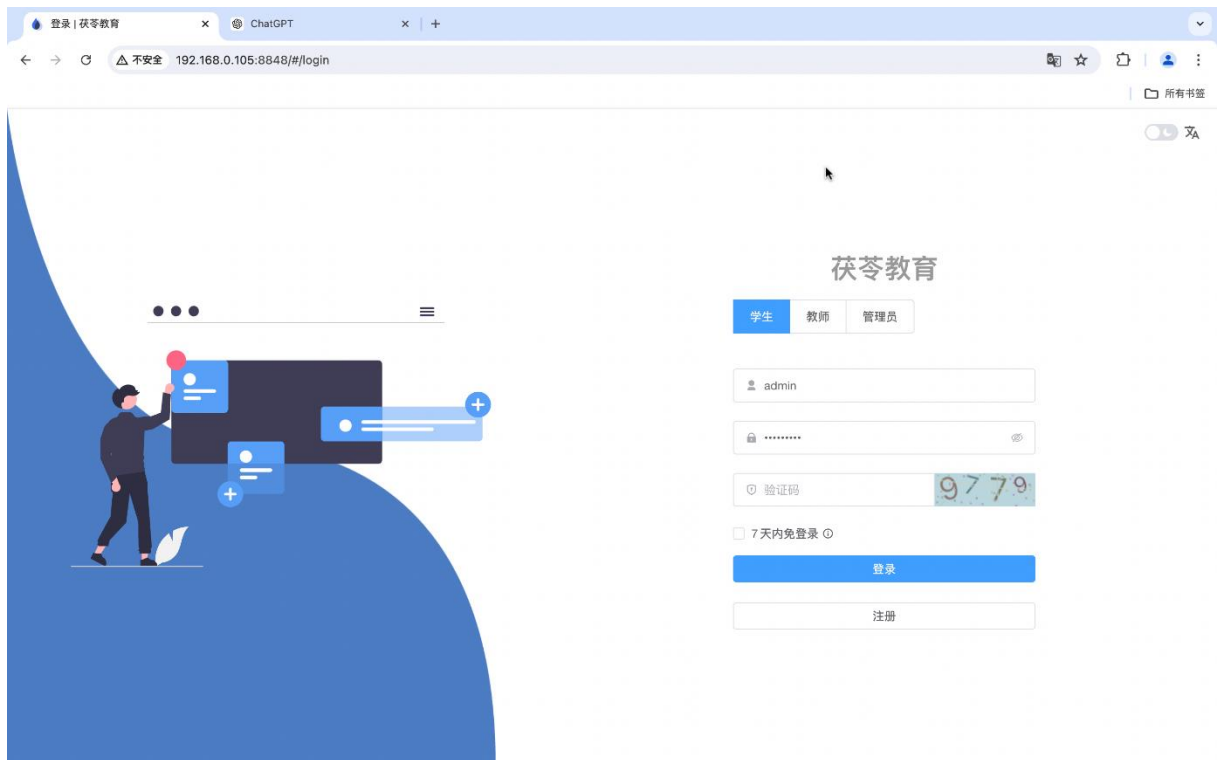


图 4.3 登录界面

4.2.2 交互逻辑

登录交互逻辑如图 4.4。

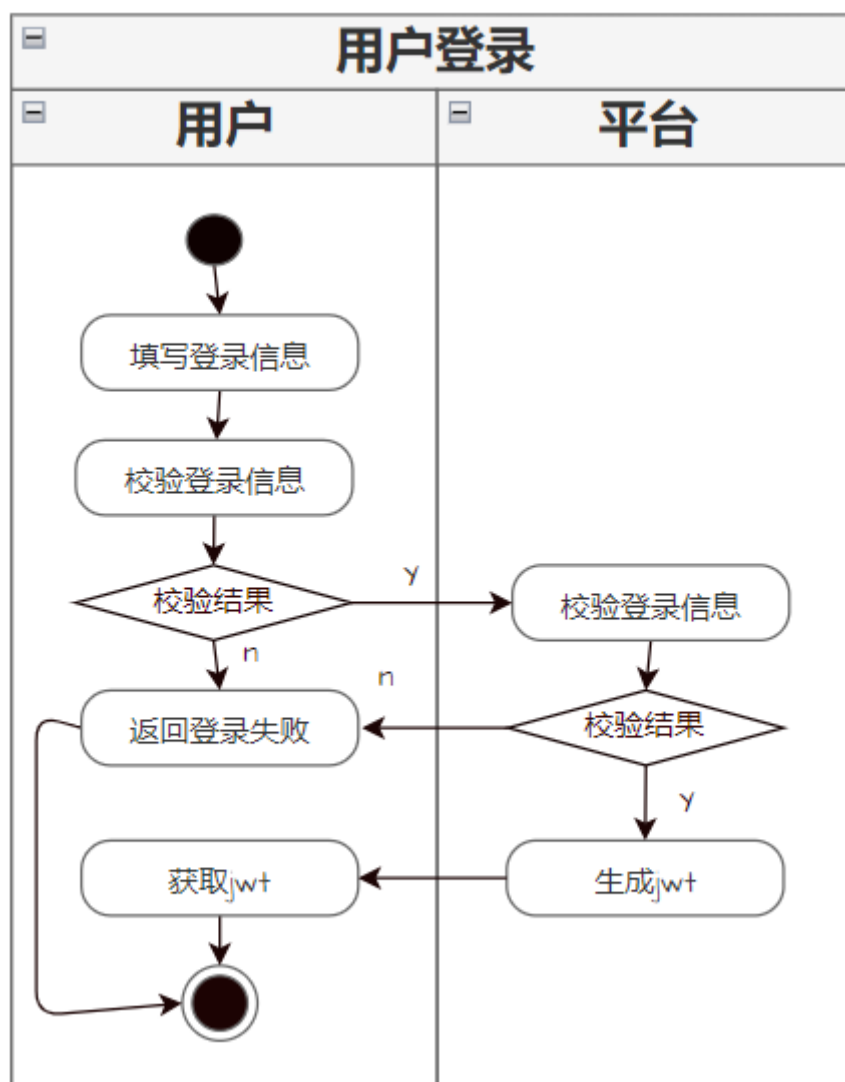


图 4.4 登录交互逻辑

4.3 教育机器人

4.3.1 界面设计

教育机器人界面设计如图 4.5。

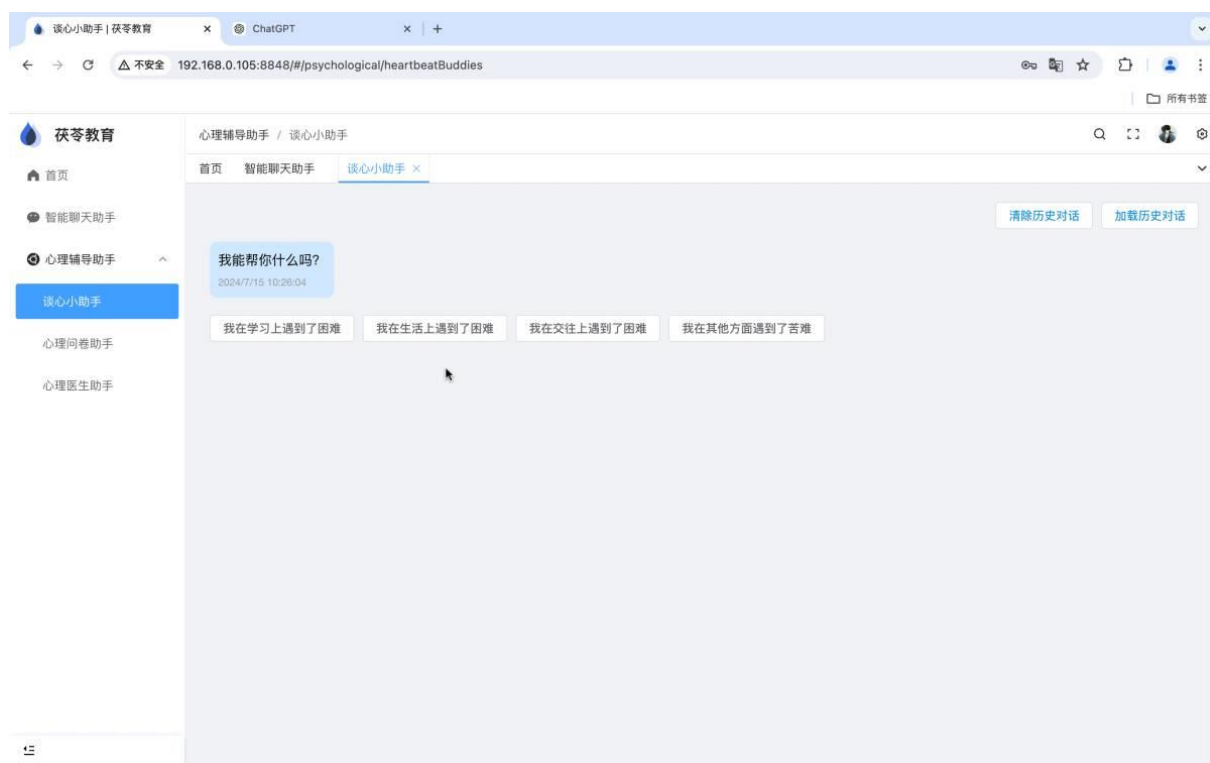


图 4.5 教育机器人界面

4.3.2 交互逻辑

教育机器人交互逻辑如图 4.6。

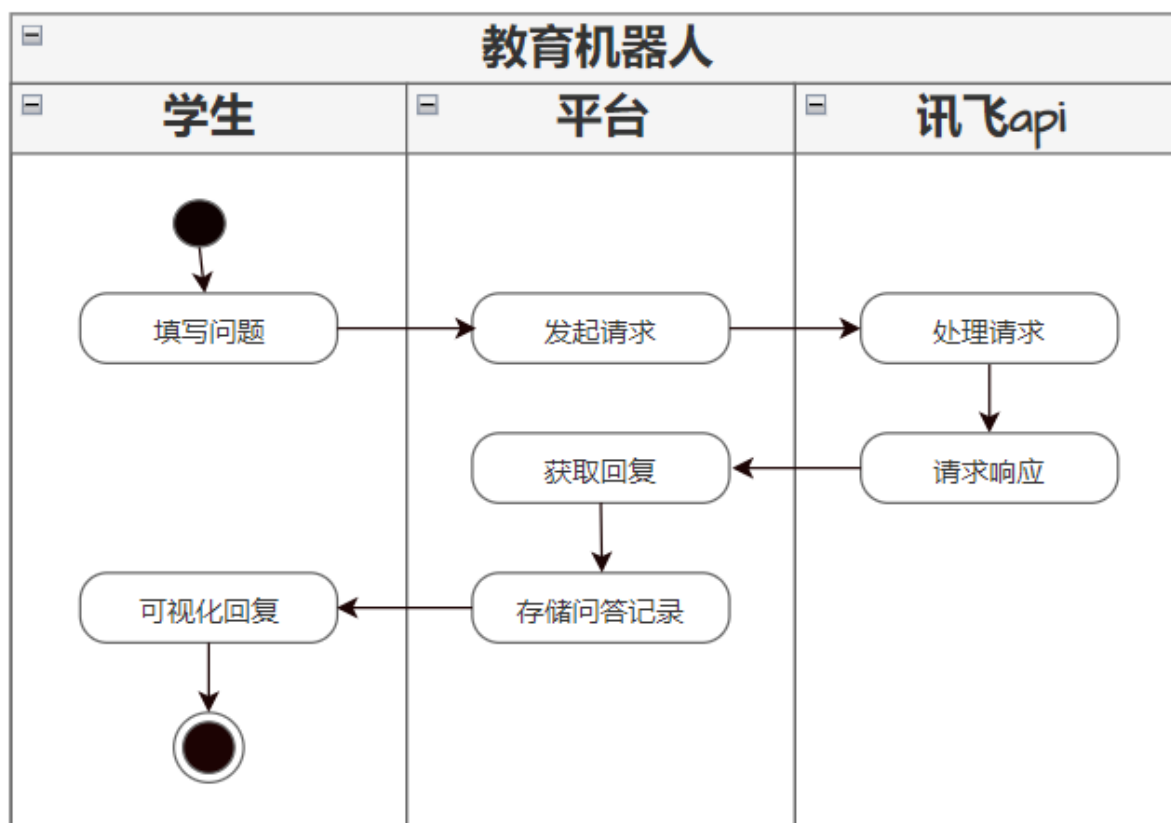


图 4.6 教育机器人交互逻辑

4.4 心理测试

4.4.1 界面设计

心理测试界面设计如图 4.7。



图 4.7 心理测试界面

4.4.2 交互逻辑

心理测试交互逻辑如图 4.8。

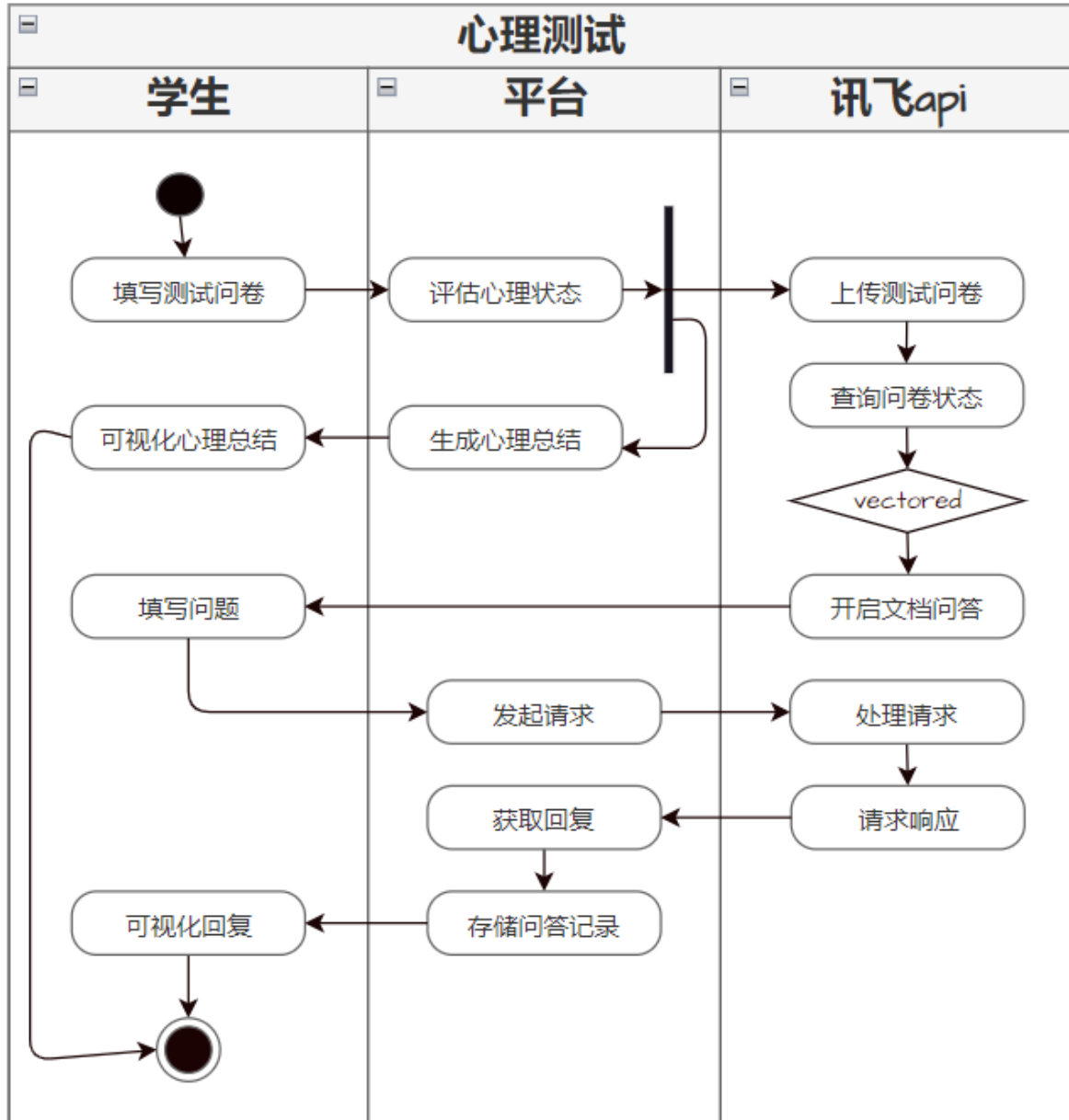


图 4.8 心理测试交互逻辑

4.5 实时英文对话

4.5.1 界面设计

实时英文对话界面设计如图 4.9。



图 4.9 实时英文对话界面

4.5.2 交互逻辑

在线英语对话交互逻辑如图 4.10。

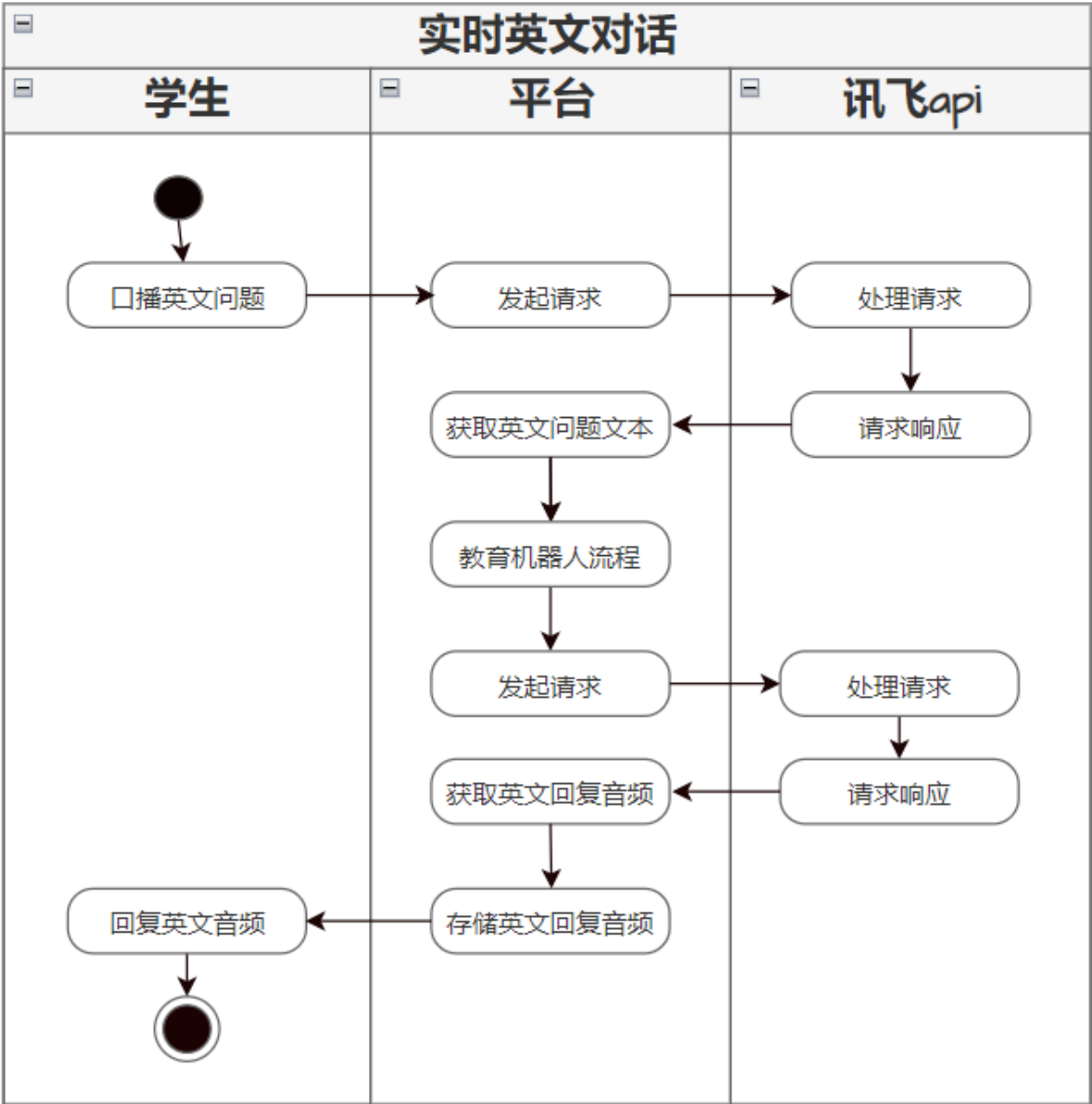


图 4.10 实时英文对话交互逻辑

4.6 教学 ppt 生成

4.6.1 界面设计

教学 ppt 生成界面设计如图 4.11。

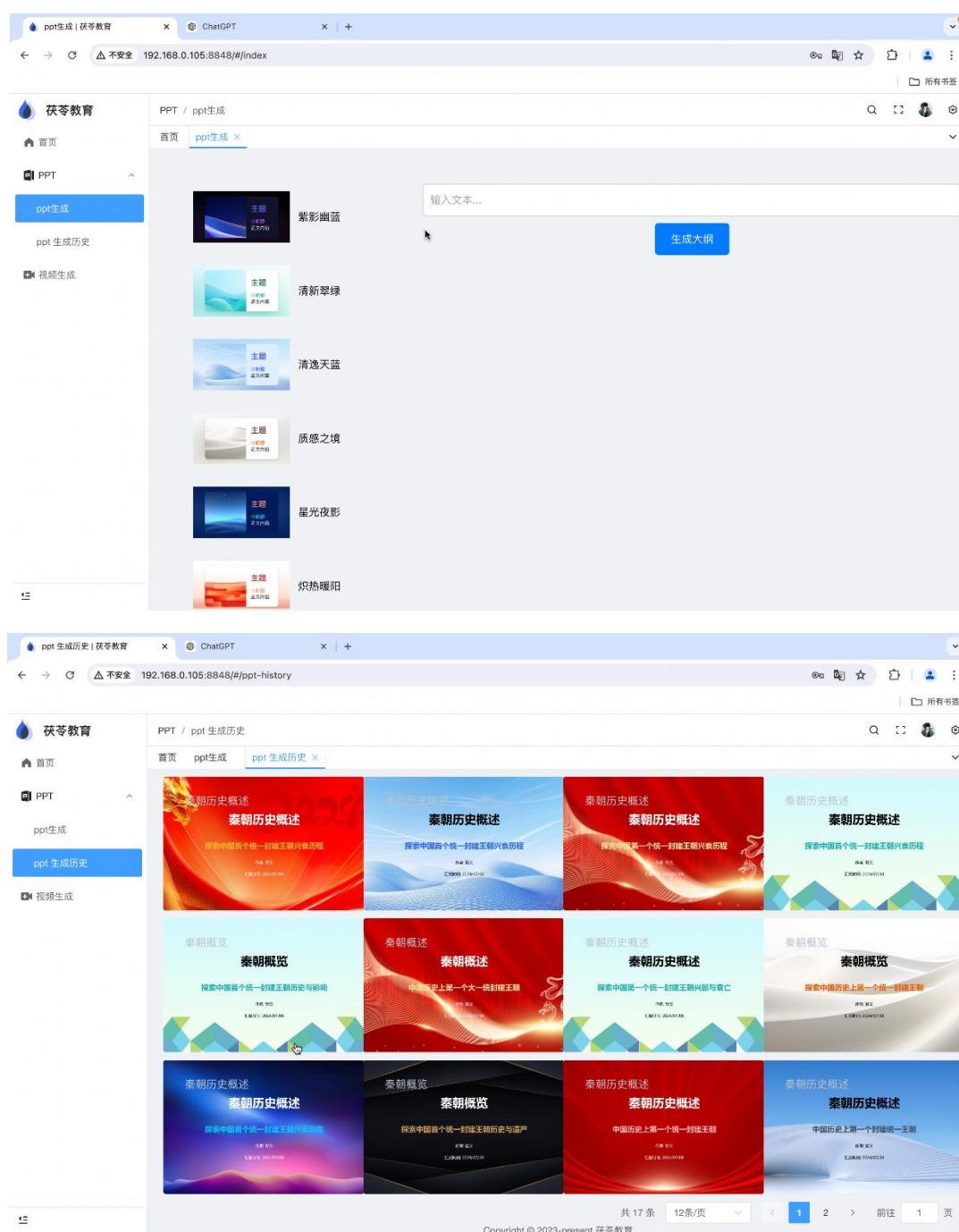


图 4.11 教学 ppt 生成界面

4.6.2 交互逻辑

教学 ppt 生成交互逻辑如图 4.12。

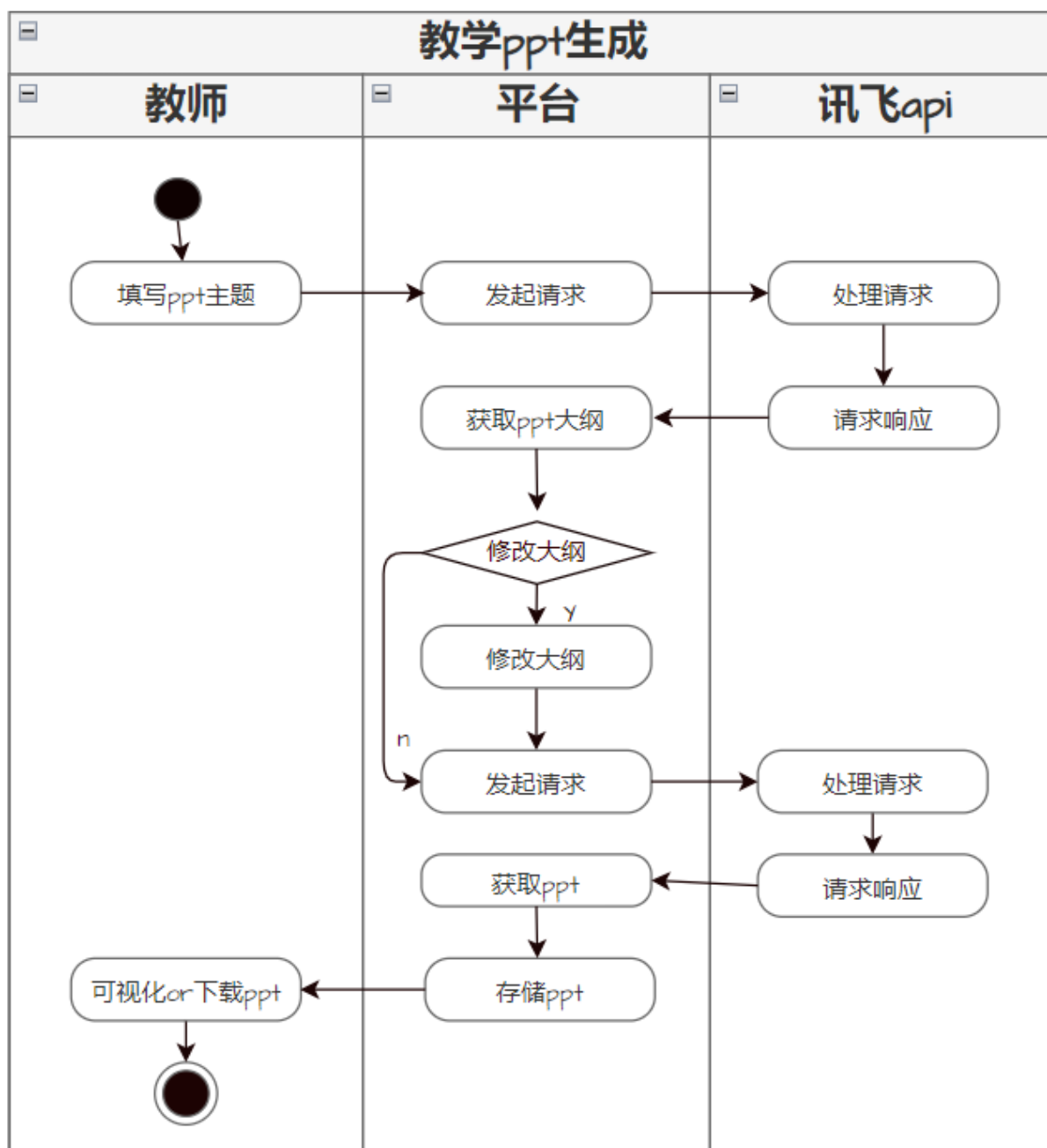


图 4.12 教学 ppt 生成交互逻辑

4.7 教学视频合成

4.7.1 界面设计

教学视频合成界面设计如图 4.13。

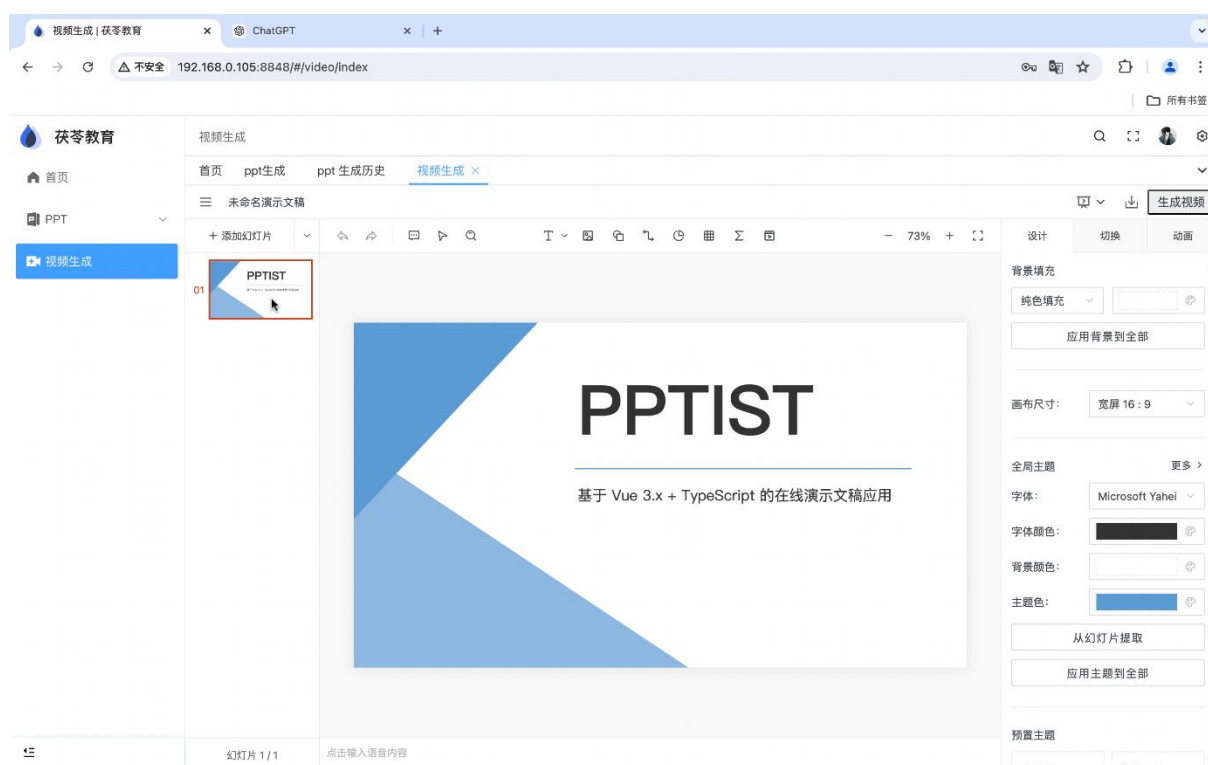


图 4.13 教学视频合成界面

4.7.2 交互逻辑

教学视频合成交互逻辑如图 4.14。

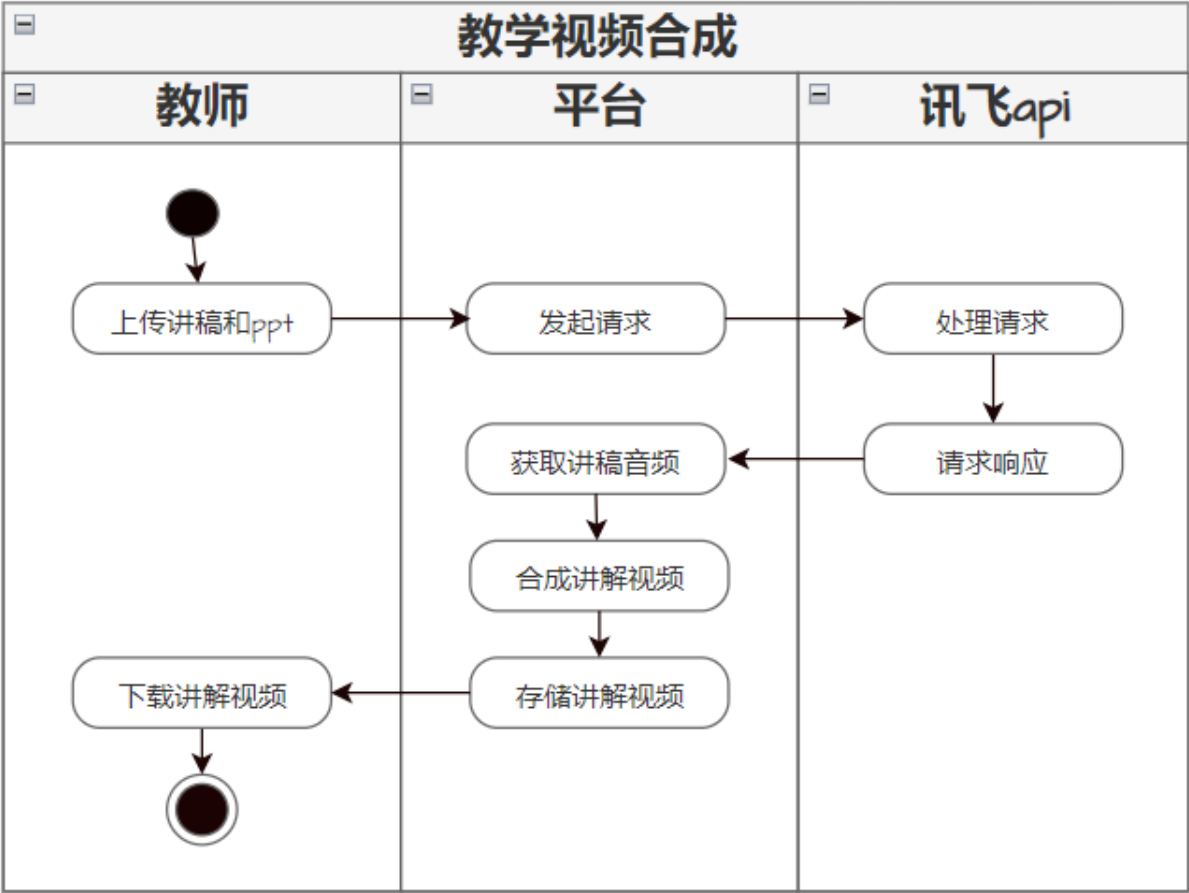


图 4.14 教学视频合成交互逻辑

4.8 用户管理

4.8.1 界面设计

用户管理界面设计如图 4.15。

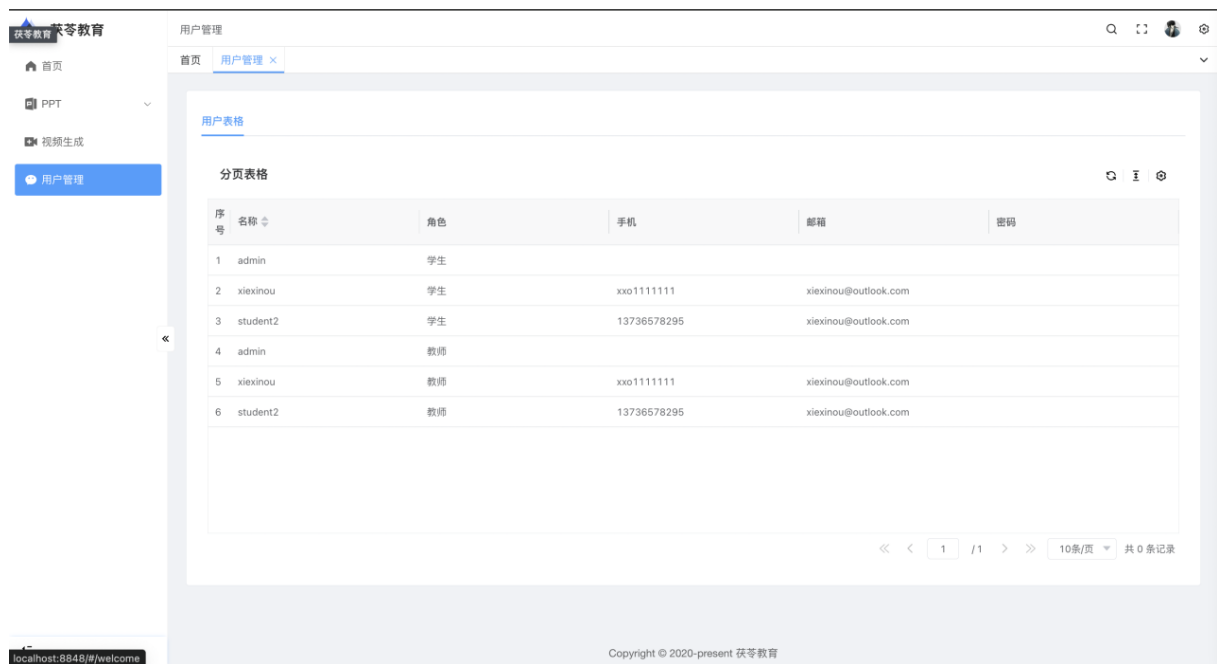


图 4.15 用户管理界面

4.8.2 交互逻辑

用户管理交互逻辑如图 4.16。

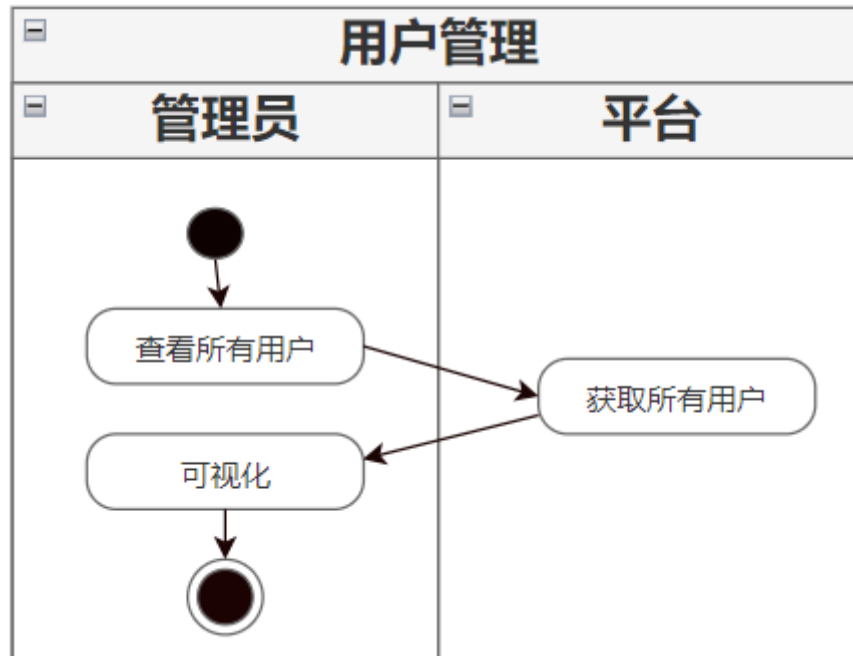


图 4.16 用户管理交互逻辑

第 5 章 核心功能设计

5.1 信息校验

用户首先需要输入信息，信息主要包括用户名，手机号，邮箱以及用户密码。然后对用户信息进行校验，校验类型包括非空校验，长度校验，格式校验以及用户名校验。对于不同的用户进行，需要进行多种不同的校验，具体校验方式如下表所示。

表 5.1 信息校验

用户信息	校验方式	说明
用户名	非空校验	若为空，返回失败
	长度校验	长度大于 10，返回失败
	用户名校验	若用户名已存在，返回失败
手机号	非空校验	若为空，返回失败
	格式校验	<code>/^1(3[0-9] 5[0-3,5-9] 7[1-3,5-8] 8[0-9])\d{8}\$/</code>
邮箱	非空校验	若为空，返回失败
	格式校验	<code>/^[a-zA-Z0-9_-]+@[a-zA-Z0-9_-]+(\.[a-zA-Z0-9_-]+)+\$/</code>
密码	非空校验	若为空，返回失败
	长度校验	长度不在 7-20 之间，返回失败
	格式校验	<code>/^[A-Za-z\d@\$!%*?&]{7,20}\$/</code>

5.2 JWT

在登录功能中，主要实现对用户登录信息实现校验，对用户的登录状态进行持久化这两个功能。其中，登录信息的校验与注册信息的校验类似，不同之处在于登录信息只包含用户名和密码。采用 jwt(JSON Web Token)实现登录状态的持久化。具体流程如下图 5.1 所示。同时采用 jwt 可以实现鉴权认证的操作，具体实现步骤是将所有请

求发送到 gateway 模块，过滤掉不含有 jwt 的请求，并在通过的请求后追加用户主键信息以及用户的类型信息，最后将请求分发到具体的服务中，如图 5.2 所示。

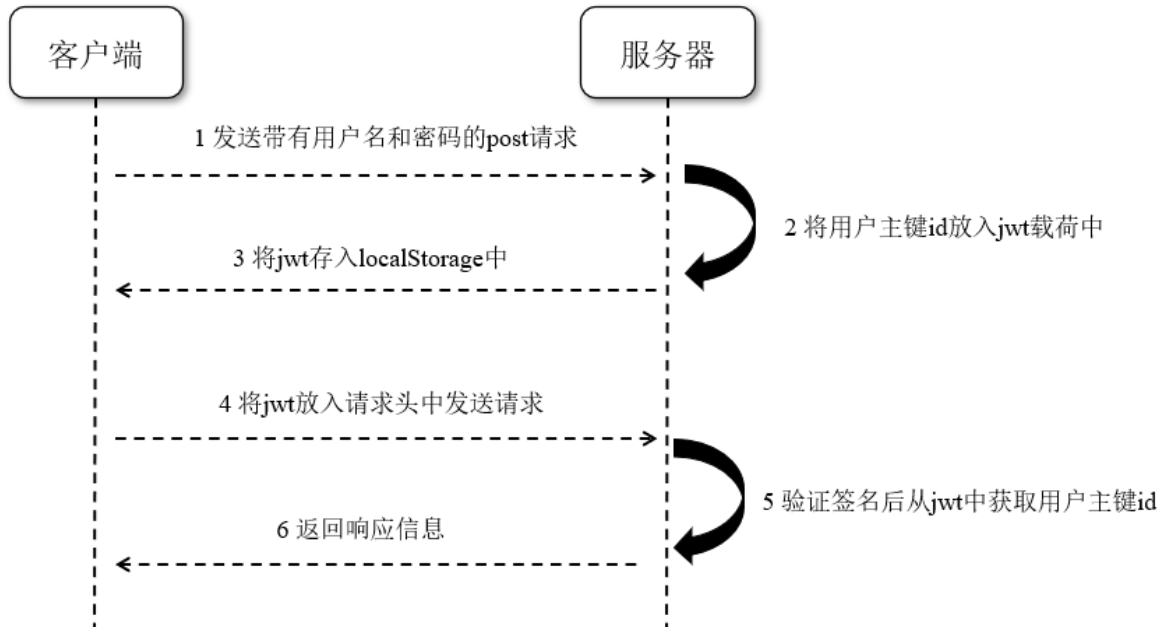


图 5.1 登录持久化流程

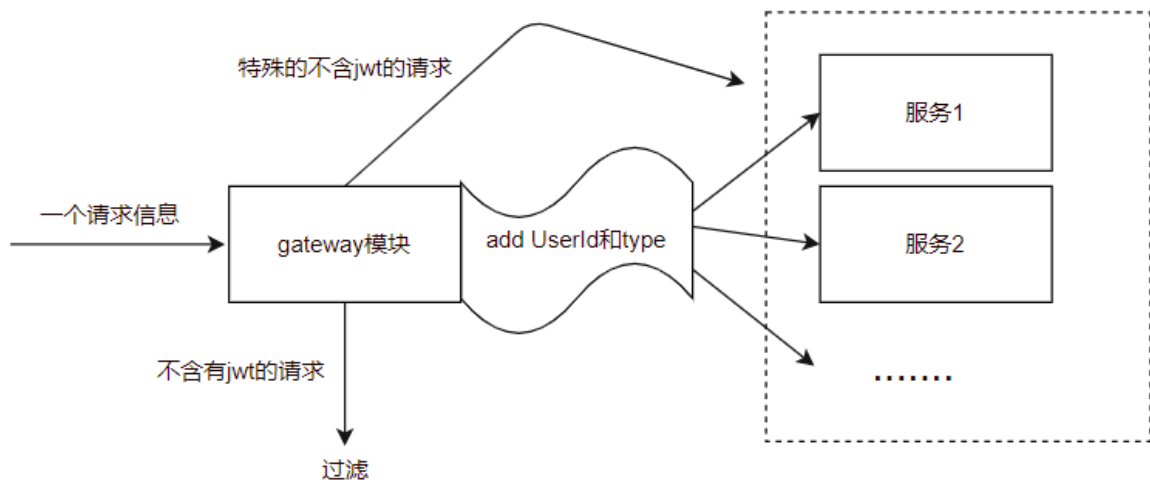


图 5.2 鉴权认证流程

5.3 信息加密

关于密码的加密，平台将其分解为密码的安全传输以及加密存储两个过程。对于密码的安全传输，过去人们常采用的方式是对称加密的方式，加密和解密双方共用一对密钥，这样加密的方式存在一个漏洞，即如果一方密钥被窃取，加密信息就会泄露。因此一种经典的加密算法被提出，即 RSA(Rivest-Shamir-Adleman)。这是一种非对称加密算法，常用于保护数据的安全传输。它基于大数因子分解的困难性问题，即给定一个大的合数，将其分解成两个素数的乘积。RSA 算法的安全性依赖于这个问题的困难性，即找到两个大素数的乘积容易，但将其分解回两个素数则极为困难。因此一方密钥的泄露不会导致密码泄露。而对于加密存储而言，哈希加密是大家通用的加密方式，这是一种单向加密，有效的防止了信息发生泄露。但对于一些简单的密码，黑客可能会通过撞库攻击进行破解。平台结合目前通用的两种解决方案，即二次哈希和加盐哈希，进行了二次哈希加盐加密，有效地解决了撞库攻击。至此完成了密码的安全传输和加密存储。

此外，对于企业而言，用户信息属于公司的机密文件。传统公司使用自增设置主键 id 值，这样竞争对手可以通过抓包获知公司的用户数，这对于公司而言是不可忍受的。因此，通过 UUID 的方法得到一个随机数来设置主键被一些公司采用，但是这种方法存在一个缺陷，即数据库表中的用户顺序不再是通过入库的先后顺序进行排序。因此平台采用由 Twitter 提出并开源的雪花算法得到主键，这样得到的用户信息会按照入库顺序进行排序，其原理在于将时间戳信息放入了主键中。

5.4 WebSocket 连接

建立 WebSocket 连接主要用于调用讯飞平台提供的需要 wss 路径的 api。WebSocket 是一种在单个 TCP 连接上进行全双工通信的协议。它设计用于在客户端（如浏览器）和服务器之间建立长时间的连接，从而减少传统 HTTP 请求中需要频繁建立和关闭连接的开销。建立连接的过程如下图 5.3 所示。



图 5.3 websocket 连接

5.5 视频合成

视频合成功能用于教学视频合成模块，该功能利用 ppt 以及讲述文稿信息合成一段教学视频，使用了开源视频合成软件 ffmpeg，具体步骤如下，用户首先上传 ppt 以及为每页附上讲述文稿，平台调用讯飞开放平台的语音合成 api 将文稿信息合成为音频信息，利用 ffmpeg 就将每一张 ppt 的图片和对应的音频合成为视频，最后将所有视频拼接就可以得到讲述视频。具体代码实现如下。

这是将图片和音频合成为视频的命令。

```
ffmpeg -loop 1 -i img1.jpg -i audiol.mp3 -c:v libx264 -tune stillimage -c:a aac -b:a 192k -pix_fmt yuv420p -shortest p -strict -2 art1.mp4。
```

这是将视频列表合成为讲述视频的命令，其中 file_list.txt 存储视频存放路径。

```
ffmpeg -f concat -safe 0 -i file_list.txt -c copy final_output.mp4
```

详细代码参考提交的代码中\back\fl edu\video-service\src\main\java\com\example\video\utils 文件中夹中的相关信息。

第 6 章 模块详细设计

6.1 用户模块

用户模块主要有 3 个功能，分别是注册、登录以及用户管理，对上述功能的描述如下表。

表 6.1 注册

用例名称	注册	
用例对象	学生、教师、管理员	
接口 1	路径	/user/exist
	输入	用户名
	输出	用户名是否存在
接口 2	路径	/user/getVerificationCode
	输入	无
	输出	向邮箱发送验证码
接口 3	路径	/user/register
	输入	用户信息类
	输出	是否注册成功
算法设计	① 调用接口 1 判断用户名是否已占用 ② 若未被占用且通过信息校验后可以调用接口 2 获取验证码 ③ 完成验证码校验后可以调用接口 3 存储注册信息 ④ 完成注册后跳转到登录界面	

表 6.2 登录

用例名称	登录	
用例对象	学生、教师、管理员	
接口 1	路径	/user/getVerificationCode
	输入	无
	输出	向邮箱发送验证码
接口 2	路径	/user/login
	输入	用户信息类
	输出	jwt 字符串
算法设计	① 调用接口 1 获取验证码 ② 完成验证码校验后可以调用接口 2 传入登录信息 ③ 如果用户名和密码正确返回 jwt 加密后的 token 字符串 ④ 完成登录后跳转到主界面	

表 6.3 用户管理

用例名称	用户管理	
用例对象	管理员	
接口 1	路径	/user/selectAllStudent
	输入	无
	输出	所有学生信息
接口 2	路径	/user/selectAllTeacher
	输入	无
	输出	所有教师信息
说明	登录管理员界面后可跳转进入用户管理，调用上述接口	

6.2 教育机器人模块

教育机器人模块核心功能在于实现类似于 chatgpt 的人机对话，通过调用讯飞开放平台星火大模型的角色模拟 api 实现，根据服务人群的的不同分化为教学助手、心理辅导和口语助手 3 个角色。对于该功能的描述如下表。

表 6.4 教育机器人

用例名称	教育机器人	
用例对象	学生	
接口 1	路径	/helper/teacherChat, /helper/mentalChat, /helper/englistChat
	输入	问题
	输出	wssURL 以及 sendText
接口 2	路径	/helper/saveChat
	输入	问题与答复
	输出	是否存储成功
接口 3	路径	/helper/getHistoryData
	输入	userId
	输出	历史问答记录
接口 4	路径	/helper/clearHistoryData
	输入	userId
	输出	是否删除成功
算法设计	① 调用接口 1 向前端提供 wssURL 以及 sendText ② 前端使用 wssURL 建立连接，通过 sendText 获取答复 ③ 调用接口 2 存储问答信息 ④ 调用接口 3 获取问答记录（可选） ⑤ 调用接口 4 删除问答记录（可选）	

6.3 心理测试模块

心理测试模块服务于心理可能存在问题的学生，该模块主要包含两个功能，分别心理测试和文档问答，后者基于测试结果展开。对上述功能的描述如下。

表 6.5 心理测试

用例名称	心理测试	
用例对象	学生	
接口 1	路径	/doc/getQuestionAndOptions
	输入	选择心理测试文档类型
	输出	心理测试文档
接口 2	路径	/doc/uploadOptions
	输入	测试结果
	输出	fileId
接口 3	路径	/doc/getConclusion
	输入	fileId
	输出	心理测试结果
算法设计	① 调用接口 1 选择一种心理测试文档 ② 填写该文档后调用接口 2 提交测试结果 ③ 完成提交后调用接口 3 进行心理总结	

表 6.6 文档问答

用例名称	文档问答	
用例对象	学生	
接口 1	路径	/doc/queryUploadProcess
	输入	fileId
	输出	是否可以进行文档问答
接口 2	路径	/doc/getWssURL, /doc/getSendText
	输入	fileId
	输出	wssURL 以及 sendText
接口 3	路径	/doc/saveChat
	输入	问题与答复、
	输出	是否存储成功
接口 4	路径	/doc/getHistoryChat
	输入	fileId
	输出	获取文档问答记录
接口 5	路径	/doc/deleteHistoryChat
	输入	fileId
	输出	删除文档问答记录
算法设计	① 调用接口 1 查看是否能够进行文档对话 ② 调用接口 2 向前端提供 wssURL 以及 sendText ③ 前端使用 wssURL 建立连接，通过 sendText 获取答复 ④ 调用接口 3 存储问答信息 ⑤ 调用接口 4 获取问答记录（可选） ⑥ 调用接口 5 删除问答记录（可选）	

6.4 实时英文对话模块

实时语音听写模块服务于需要英语口语练习的学生，用户开始对话后可以使用英语提出问题，平台使用英语答复，对话记录显示在屏幕上。该模块主要由三部分构成，分别是语音听写，语音合成以及教育机器人模块中的口语助手，下表主要描述前两个功能。

表 6.7 语音听写

用例名称	语音听写	
用例对象	学生	
接口 1	路径	/virtual/getListenWssUrl
	输入	无
	输出	wssURL
接口 2	路径	/virtual/getListenRequest
	输入	无
	输出	讯飞 api 请求体
算法设计	① 调用接口 1 得到语音听写 wssURL ② 建立 wss 连接并接收音频信息 ③ 调用接口 2 获得讯飞语音听写 api 请求体，将音频信息放入请求体中，发出请求，获取识别文本 ④ 将识别文本提供给口语助手，等待回复	

表 6.8 语音合成

用例名称	语音听写	
用例对象	学生	
接口 1	路径	/virtual/getSpeakWssUrl
	输入	无
	输出	wssURL
接口 2	路径	/virtual/getSpeakRequest
	输入	无
	输出	讯飞 api 请求体
算法设计	① 调用接口 1 得到语音听写 wssURL ② 建立 wss 连接并获取口语助手的回复 ③ 调用接口 2 获得讯飞语音听写 api 请求体，将回复信息放入请求体中，发出请求，获取回复音频 ④ 播放回复音频	

6.5 教学 ppt 生成模块

教学 ppt 生成模块服务于教师，教师可以通过输入一段主题获取到与之相关的大纲，再通过大纲获取到生成的 ppt。该模块分为大纲生成和 ppt 生成两个功能。下表对上述功能的描述如下。

表 6.9 大纲生成

用例名称	大纲生成	
用例对象	教师	
接口 1	路径	/ppt/createOutline
	输入	一段主题文字
	输出	fileId
接口 2	路径	/ppt/updateOutline
	输入	fileId, 更新后大纲
	输出	fileId
接口 3	路径	/ppt/getOutline
	输入	fileId
	输出	大纲
接口 4	路径	/ppt/getThemeList
	输入	无
	输出	ppt 背景列表
算法设计	① 调用接口 4 选择 ppt 生成背景，调用接口 1 生成大纲 ② 调用接口 2 更新大纲（可选） ③ 调用接口 3 得到大纲	

表 6.10 ppt 生成

用例名称	ppt 生成	
用例对象	教师	
接口 1	路径	/ppt/createPPT
	输入	fileId
	输出	fileId
接口 2	路径	/ppt/queryPPT
	输入	fileId
	输出	进度信息
接口 3	路径	/ppt/getPPT
	输入	fileId
	输出	生成的 ppt 文件
接口 4	路径	/ppt/getHistoryData
	输入	userId
	输出	历史生成 ppt 的 fileId 列表
接口 5	路径	/getCoverImgSrc、/ppt/getName/、 /ppt/getTimeLastChange
	输入	fileId
	输出	ppt 生成过程中产生的相关信息
算法设计	① 调用接口 1 创建 ppt 生成任务 ② 调用接口 2 查询 ppt 生成任务进度 ③ 成功生成 ppt 后可以调用接口 3 获取 ppt 文件 ④ 调用接口 4 获取历史生成的 ppt 的 fileId 列表（可选） ⑤ 调用接口 5 获取生成 ppt 的相关信息（可选）	

6.6 教学视频合成模块

自线上教育而来，越来越多的教师采用录课的方式将自己的讲课内容记录并发表在网络上。而需要录制课程的教师往往会花费不少的时间、金钱和精力在上面。教学视频合成模块可以大大提高这一过程的效率。该模块需要提供 ppt 以及为每页 ppt 附上讲述文本，其功能主要分为合成讲述音频以及合成视频两部分。具体描述如下。

表 6.11 讲述音频合成

用例名称	讲述音频合成	
用例对象	教师	
接口 1	路径	/video/uploadPPT
	输入	上传 ppt
	输出	fileId
接口 2	路径	/video/uploadTextList
	输入	上传讲述文本列表，fileId
	输出	fileId
接口 3	路径	/video/queryMp3Process
	输入	fileId
	输出	音频生成进度
算法设计	① 首先调用接口 1 创建任务，将 ppt 传到后端并读取成图片 ② 调用接口 2 创建音频生成任务 ③ 调用接口 3 查询音频生成进度	

表 6.12 视频合成

用例名称	视频合成	
用例对象	教师	
接口 1	路径	/video/createVideo
	输入	fileId
	输出	fileId
接口 2	路径	/video/getInfo
	输入	fileId
	输出	视频生成任务进度信息
接口 3	路径	/video/mp4/{fileId}
	输入	fileId
	输出	生成的视频文件
接口 4	路径	/video/pptx/{fileId}
	输入	fileId
	输出	上传的 ppt 文件
接口 5	路径	/video/getHistoryData
	输入	userId
	输出	历史生成的视频的 fileId 信息
算法设计	① 调用接口 1 创建视频生成任务 ② 调用接口 2 查询视频生成任务进度 ③ 成功生成视频后可以调用接口 3 获取视频文件 ④ 成功上传 ppt 后可以调用接口 4 获取 ppt 文件（可选） ⑤ 调用接口 5 获取历史生成的 ppt 的 fileId 列表（可选）	

第 7 章 技术设计

7.1 技术栈

本平台采用的技术栈如下图所示。其中前端主要采用 vue3 构建应用，使用 axios 用于发起 http 请求和处理响应数据，使用 ElementUI 构建界面交互，使用 webpack 进行打包，使用 node.js 构建运行环境，使用 nginx 进行服务的注册与监听。后端采用 gateway 作为网关统一处理前端的请求，并进行负载均衡和鉴权认证，使用 JWT 合成鉴权，使用 Java1.8 构建运行环境，使用 mysql 存储数据，使用 nacos 进行服务注册，使用 ffmpeg 进行视频相关操作，使用 swagger 辅助前端开发，使用 feign 调用讯飞开放平台的第三方 api。最后，讯飞开放平台为本平台提供包括但不限于角色模拟，ppt 生成、语音合成、语音听写，文档问答相关 api 的支持。



图 7.1 技术栈